NAVAL POSTGRADUATE SCHOOL

Monterey, California

THESIS

PRELIMINARY HELICOPTER DESIGN DECISION MAKING BASED ON FLIGHT PERFORMANCE FACTORS

by

Patrick V. Adamcik

September 1984

Thesis Advisor: Donald M. Layton

Approved for public release: Distribution unlimited

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

Field	Content
1. REPORT NUMBER	
2. GOVT ACCESSION NO.	
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle)	Preliminary Helicopter Design Decision Making Based on Flight Performance Factors
5. TYPE OF REPORT & PERIOD COVERED	Master's Thesis September 1984
6. PERFORMING ORG. REPORT NUMBER	
7. AUTHOR(s)	Patrick V. Adamcik
8. CONTRACT OR GRANT NUMBER(s)	
9. PERFORMING ORGANIZATION NAME AND ADDRESS	Naval Postgraduate School Monterey, California 93943
10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS	
11. CONTROLLING OFFICE NAME AND ADDRESS	Naval Postgraduate School Monterey, California 93943
12. REPORT DATE	September 1984
13. NUMBER OF PAGES	97
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	
15. SECURITY CLASS. (of this report)	Unclassified
15a. DECLASSIFICATION/DOWNGRADING SCHEDULE	

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release, distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Helicopter Design, Helicopter Performance, Helicopter Flight Performance, Helicopter Performance Factors

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This thesis will assist those evaluating helicopter design to make preliminary judgments about the feasibility of new designs. By using the computer program developed in this thesis, a designer can produce estimates for power requirements, endurance velocity, rate of climb, range velocity, hover ceiling, and service ceiling versus main rotor radius. These

20. continuation

estimates can also be examined for the effects of changes in main rotor radius, chord, and rotational velocity.

Approved for public release; distribution unlimited

Preliminary Helicopter Design Decision Making Based on Flight Performance Factors

by

Patrick V. Adamcik
Captain, United States Army
B.S., University of Texas at Austin, 1977

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN AERONAUTICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
September 1984

ABSTRACT

This thesis will assist those evaluating helicopter design to make preliminary judgments about the feasibility of new designs. By using the computer program developed in this thesis, a designer can produce estimates for power requirements, endurance velocity, rate of climb, range velocity, hover ceiling, and service ceiling versus main rotor radius. These estimates can also be examined for the effects of changes in main rotor radius, chord, and rotational velocity.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGMENT

Acknowledgment is made of the assistance provided by Professor Donald M. Layton, thesis advisor , who helped in guiding the direction of this thesis. Much of the necessary information for this thesis was found during Professor Layton's exellent classes on performance and design of helicopters.

I. INTRODUCTION

A military engineering officer in a program or project office would probably never be called on to prepare a conceptual design of a helicopter, but he may well be required to evaluate a proposal submitted by a commercial contractor.

The requirement for such an evaluation might be stated as:

> Determine if this design meets or exceeds the performance factors listed on the next page and if the design can be changed to optimize the performance.

There are several approaches one could use to complete this task. One could use a hand held calculator and enter the data for the many equations, or one could write a program for a micro or main frame computer to produce the required information. Both these options are laborious and time-consuming. By choosing the first option, one may be able to complete the task in two or three weeks or if it is decided to write a program, one may finish in one week provided that there are no "bugs".

The best solution of course, is to use an existing program which can determine the necessary information for the specification parameters. In addition, this program should be able to show what happens if certain parameters

are allowed to vary, thus providing the formation for optimization.

The objective of this thesis was to develop such a program that will graphically represent power requirements, endurance velocity, rate of climb, range velocity, hover ceiling and service ceiling all as a function of main rotor radius for four different cases. The graphs that result from this program should enable one not only to evaluate the basic helicopter design, but also to make recommendations for design improvement.

II. APPROACH TO THE PROBLEM

Six performance factors (power requirements, endurance velocity, rate of climb, range velocity, hover ceiling and service ceiling) were determined to be major in designing of a helicopter. In order to obtain estimates for these, a Fortran program was to be written to be used with "DISSPLA" (Display Integrated Software System and Plotting Language),[Ref. 1], to plot the results of the performance factors versus main rotor radius on either a screen or as a hard copy from the main frame computer.

The equations required for calculating these factors were obtained from "Helicopter Performance",[Ref. 2], and "Helicopter Design Manual",[Ref. 3], both written by Donald M. Layton. These equations were examined and then grouped for the purpose of writing subroutines to ensure an effective use of computer time. The result was six subroutines, Sub-group A, [Appendix B], are used in all the performance calculations. Sub-group A consist of:

1. RHO - Calculates density from pressure altitude and temperature or given density altitude. Also determines pressure and temperature ratio for altitude versus sea level.

2. VELMR - Calculates induced velocity for main rotor.

3. VELTR - Calculates induced velocity for tail rotor.

4. POWMR - Calculates power requirements for main rotor only.

5. POWTR - Calculates power requirements for tail rotor and total aircraft.

6. CONST - Calculates general constants to be used in the first five subroutines listed above.

In addition to the subroutines listed above, six more, Sub-group B, [Appendix B], were written to perform the necessary iterations to calculate the performance factors. Some of Sub-group B subroutines were iterated in steps (1000,100,10,1) so that it would not require extensive computer time to obtain the desired results. A final eight subroutines, Sub-group C, [Appendix B], were written to plot the results using DISSPLA.

The program takes into account neither compressibility nor blade stall. These considerations were omitted since this program was aimed at facilitating preliminary decision making for a new design, rather than a final production design, and usually the blade selected for a new design is taken from an existing helicopter.

III. SOLUTION TO THE PROBLEM

The program requires input data prior to compiling and execution. To demonstrate how to use the program a set of sample data was used. Table 1 is a listing of the required input data for the program.

Table 1

Sample Data for Use in Program

	Main Rotor	Tail Rotor
Radius	26.8 ft	5.5 ft
Chord	1.75 ft	0.81 ft
Rotational velocity	27.0 rad/sec	124.6 rad/sec
Coefficient of drag	0.008	0.008
Number of blades	4	4

General Helicopter Data

Weight	20000.0 lbs
Tail boom length	31.5 ft
Effective flat plate area (forward)	25.7 sq ft
Effective flat plate area (vertical)	31.8 sq ft
Main rotor height above skids or wheels	11.2 ft
Aircraft velocity (power calculations only)	90.0 kts
Plus/minus (+/-) value for main rotor radius	4.0 ft

Engine Data

Number of engines	2
Shaft horsepower output (military)	1561.0 shp
Shaft horsepower output (normal)	1318.0 shp
Shaft horsepower output (cruise)	989.0 shp
Specific fuel consumption (military)	0.46 lbs/shp/hr
Specific fuel consumption (normal)	0.47 lbs/shp/hr
Specific fuel consumption (cruise)	0.51 lbs/shp/hr

As indicated in Table 1, a value is inputted for the range over which you wish to examine the main rotor radius.

The example uses a value of four (4.0) feet. This value is assigned to a variable called "diff". The program was written to allow the X-axis scale of the plot to be adjusted in accordance with any value you might choose for "diff". The Y-axis scales can also change based on the maximum and minimum values for the performance factor being considered.

Certain assumptions had to be made in order to write the program. Table 2 lists these assumptions and factors used and the line number in the program where the factors can be changed if needed.

Table 2

Assumptions Made in the Program

Assumption	Factor	Line Number
Single Mast Helicopters	NA	NA
Only Rectangular Blades	NA	NA
Profile Power Factor	4.3	1079,1108
Ground Effect Factor	1.6	1138
Transmission & Accessory Losses		
1 engine	SHP = (# ENG*ESHP-10.0)*0.97	560,780,887
2 engines	SHP = (# ENG*ESHP-10.0)*0.9*0.97	561,781,888
3 engines	SHP = (# ENG*ESHP-10.0)*0.9*0.94	562,782,889

ENG: Engine, ESHP: Engine shaft horsepower,
SHP: Rotor Shaft horsepower

The Transmission and Accessory Losses were calculated based on a loss of ten (10) horsepower for the accessories, ten percent (10%) for multiple engine installation, and three

percent (3%) for transmission losses (if three engines are used, an additional transmission is required), [Ref. 2].

The secondary goal of the thesis was to examine four (4) different cases involving main rotor Radius, Chord, Rotational Velocity, Tip Velocity, Advance Ratio, and Solidity. Table 3 shows how these variables were used in each of the four (4) cases.

Table 3

Case Use of Variables

	Radius	Chord	Rotational Velocity	Advance Ratio	Tip Velocity	Solidity
Case 1	V	C	C	V	V	V
Case 2	V	C	V	C	C	V
Case 3	V	V	C	V	V	C
Case 4	V	V	V	C	C	C

V : VARIES   C : HELD CONSTANT

Advance Ratio, Tip Velocity, and Solidity when held constant keep their values at the specification conditions. The values of Chord and Rotational Velocity were allowed to vary when Advance Ratio, Tip Velocity, or Solidity were held constant as shown in Table 3.

The graphs consist of two (2) plots representing the same data but presented in different ways. The left hand plot shows the actual performance factor value versus main rotor radius. Figure 3.1 shows an example using power required versus main rotor radius, Case 1.

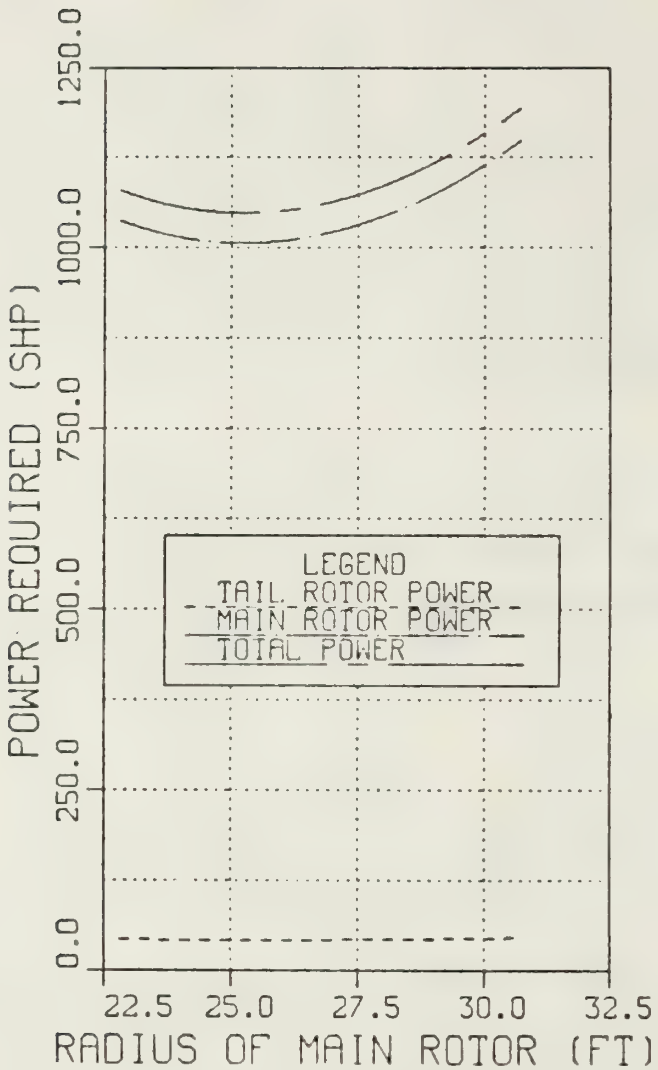


Figure 3.1 Example of Left Hand Plot

The right hand plot shows the values of the performance factors based on a percent difference from the value at the specification conditions versus a percent difference of the radius from the radius specification. Figure 3.2 shows an example of power required (% change) versus radius of the main rotor (% change), Case 1.

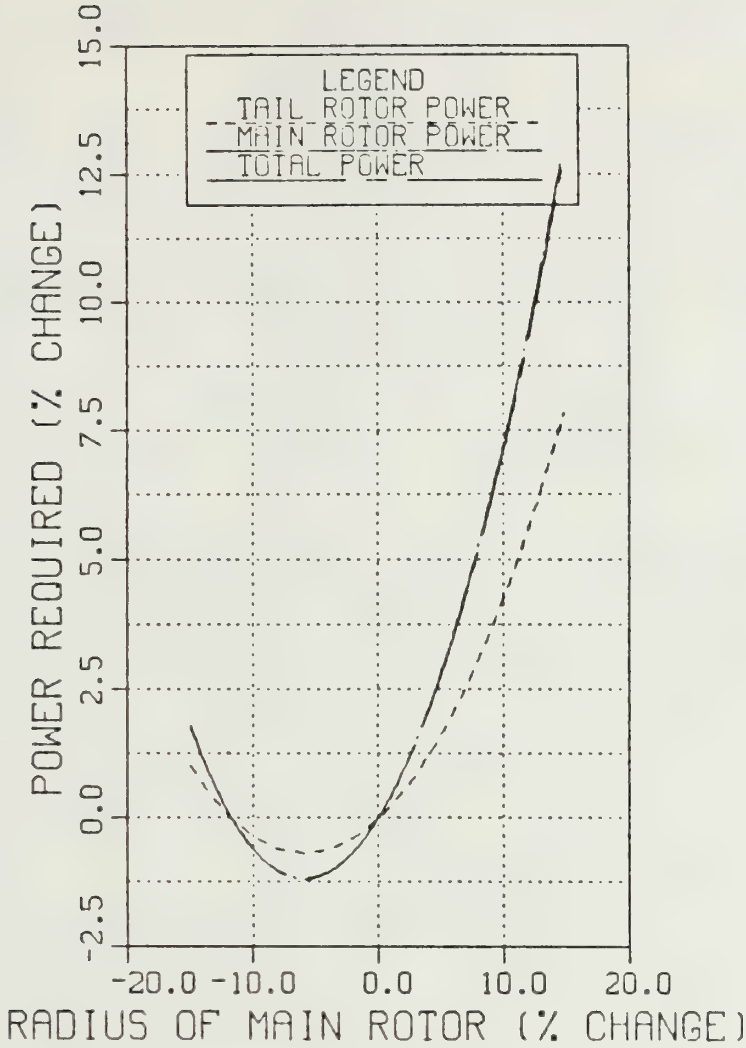


Figure 3.2 Example of Right Hand Plot

IV. RESULTS

The program will generate a total of thirty two (32) plots. The first twelve (12) pertain to power requirements. This includes induced, profile, and parasite power for the main rotor; and induced and profile power for the tail rotor. To indicate a relation between the power terms of the main rotor the figure of merit (induced divided by the sum of the induced and profile power) is included. The sum of the terms (total power) for the main and tail rotor was included with the total power required for the aircraft. The remaining twenty (20) plots are for the performance factors of endurance velocity, rate of climb, range velocity, hover ceiling and service ceiling.

The data for this particular model (UH-60A) was used to illustrate the features of the program and the results presented are typical for this example only. The specifications of any model or design can be used as long as the assumption of a single-mast helicopter with rectangular blades is followed.

The plots created by the program represent the results of varying the main rotor radius, chord or rotational velocity. To example how to interrupt the results, the four (4) plots for Maximum Rate of Climb will be analyzed. Table 4 compares the results in relation to rate of climb, radius

and percent increase of rate of climb. The cases were explained in Table 3 and the Figures follow Table 4.

Table 4

Case Evaluation of Maximum Rate of Climb Plots

Case	Figure Reference	Rate of Climb(ft/sec)	Radius (ft)	Percent Increase
1	4.1	2628	27.8	+ 0.130
2	4.2	2668	24.8	− 0.156
3	4.3	2700	30.63	+ 2.80
4	4.4	2637	25.63	− 5.00

From examining Table 4, it would be best to increase the radius to 30.63 feet while holding the chord to 1.75 feet. the tip velocity to 723.6 ft/sec (radius times rotational velocity), changing the rotational velocity to 23.62 rad/sec, and calculating solidity from radius and chord. This, however, may not be the best answer if there is a limitation to the radius of the blade. For this particular example, the optimum solution may be to leave the design as it is. Increasing or decreasing the radius by small amounts does little to increase or decrease the rate of climb.

One must not only look at one factor in making the final decision. The changes affect the performance factors in different ways, therefore, the final decision must be made after weighting all the changes against all the performance factors.

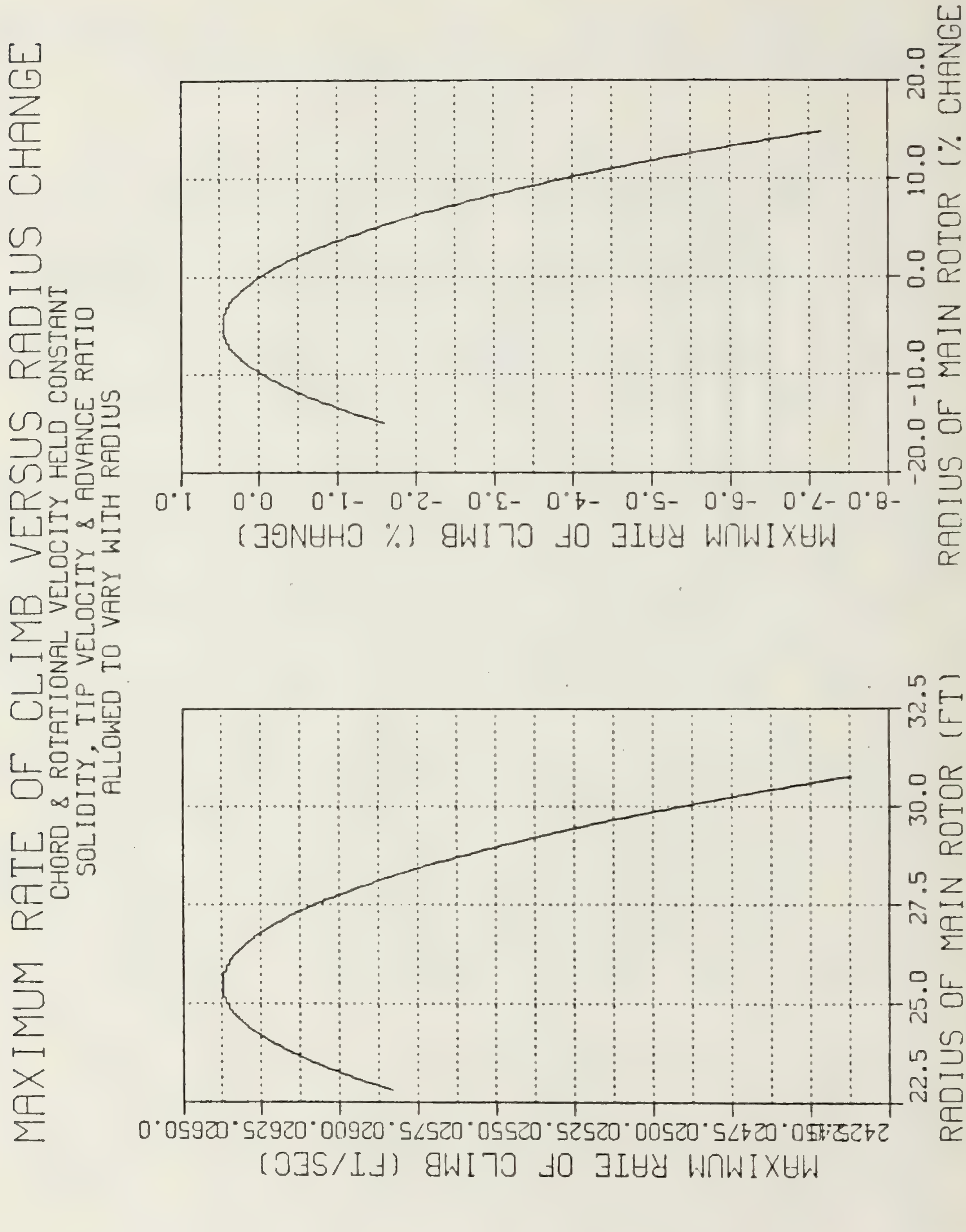


Figure 4.1 Example Maximum Rate of Climb - Case 1

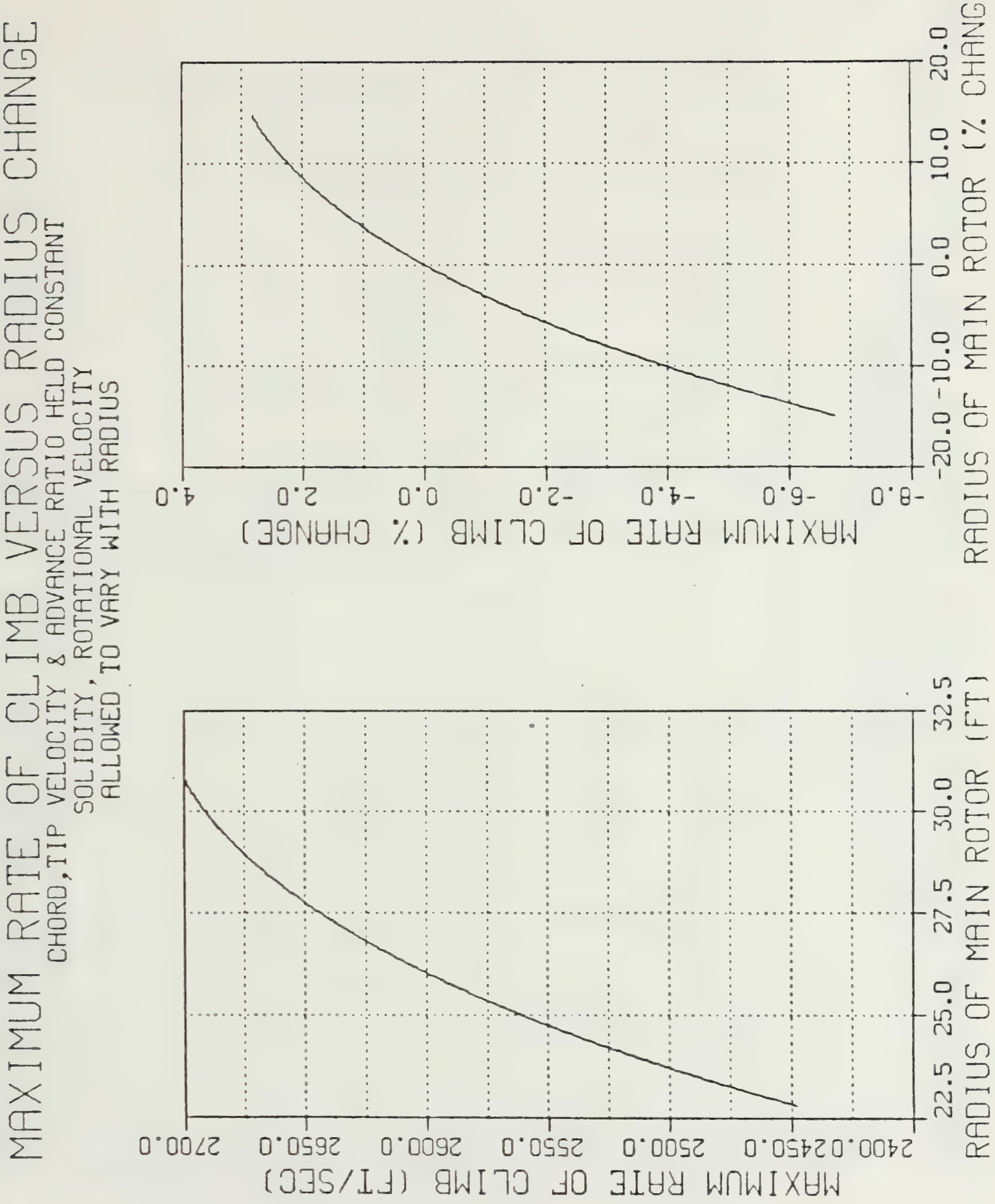


Figure 4.2 Example Maximum Rate of Climb - Case 2

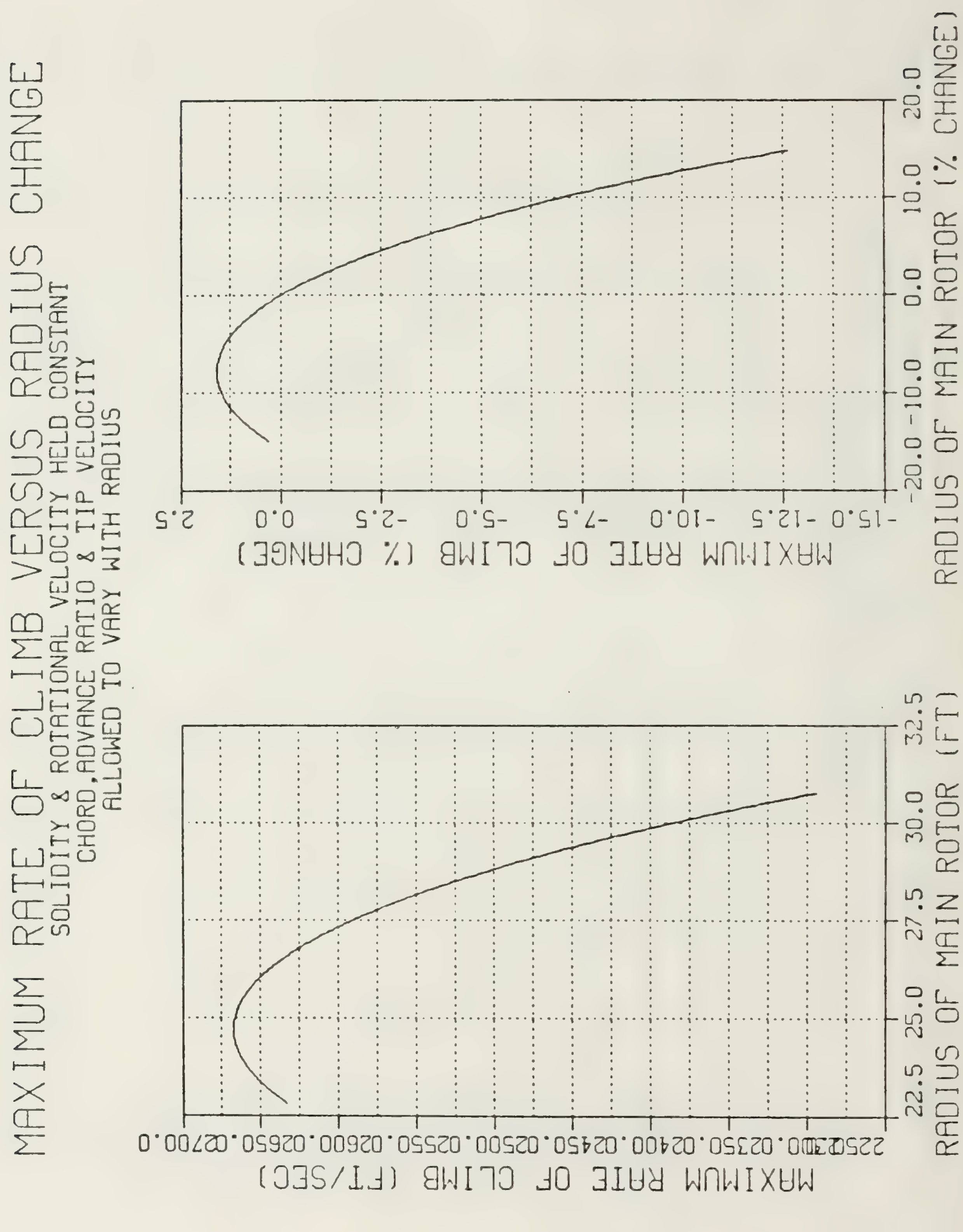


Figure 4.3 Example Maximum Rate of Climb - Case 3

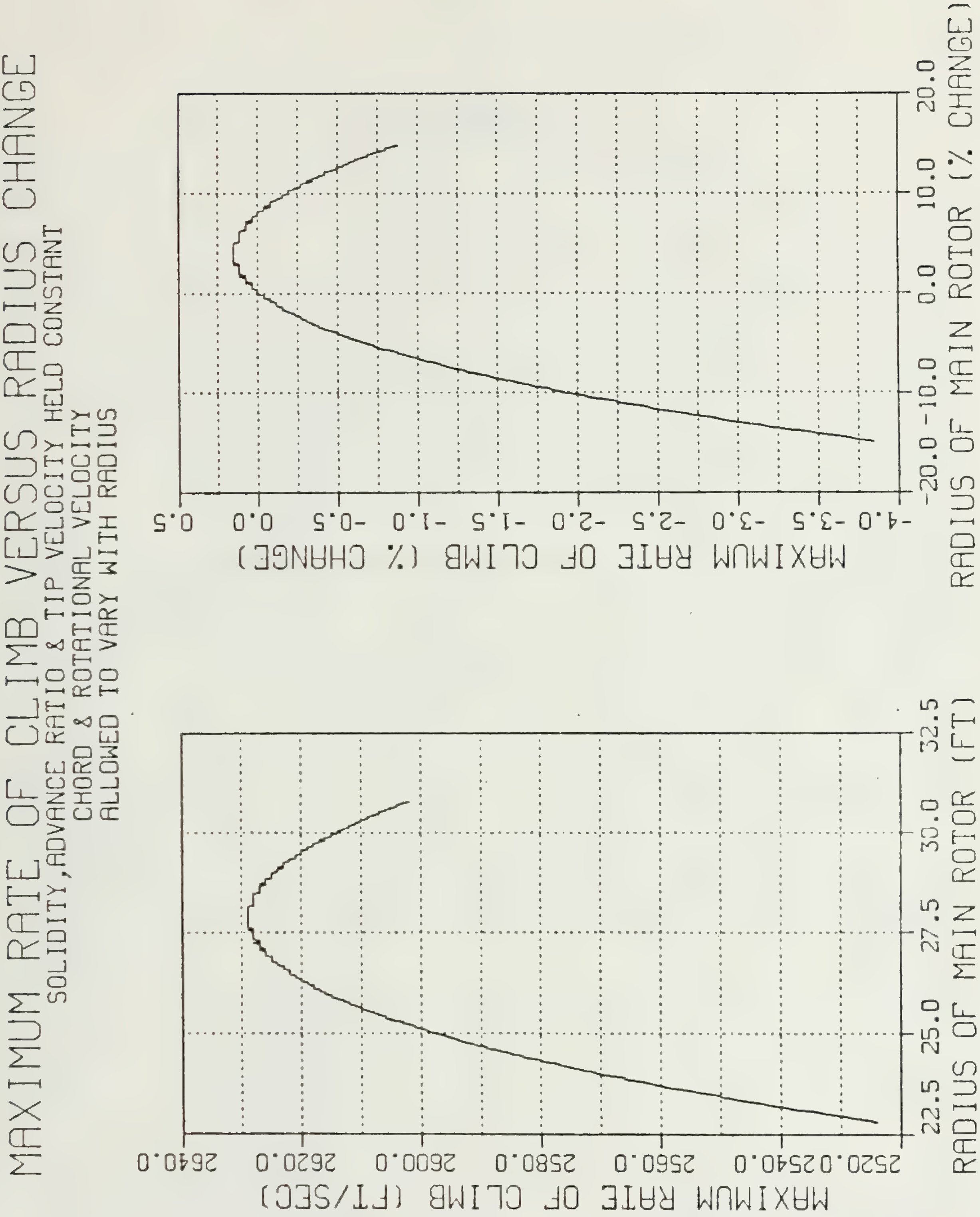


Figure 4.4 Example Maximum Rate of Climb - Case 4

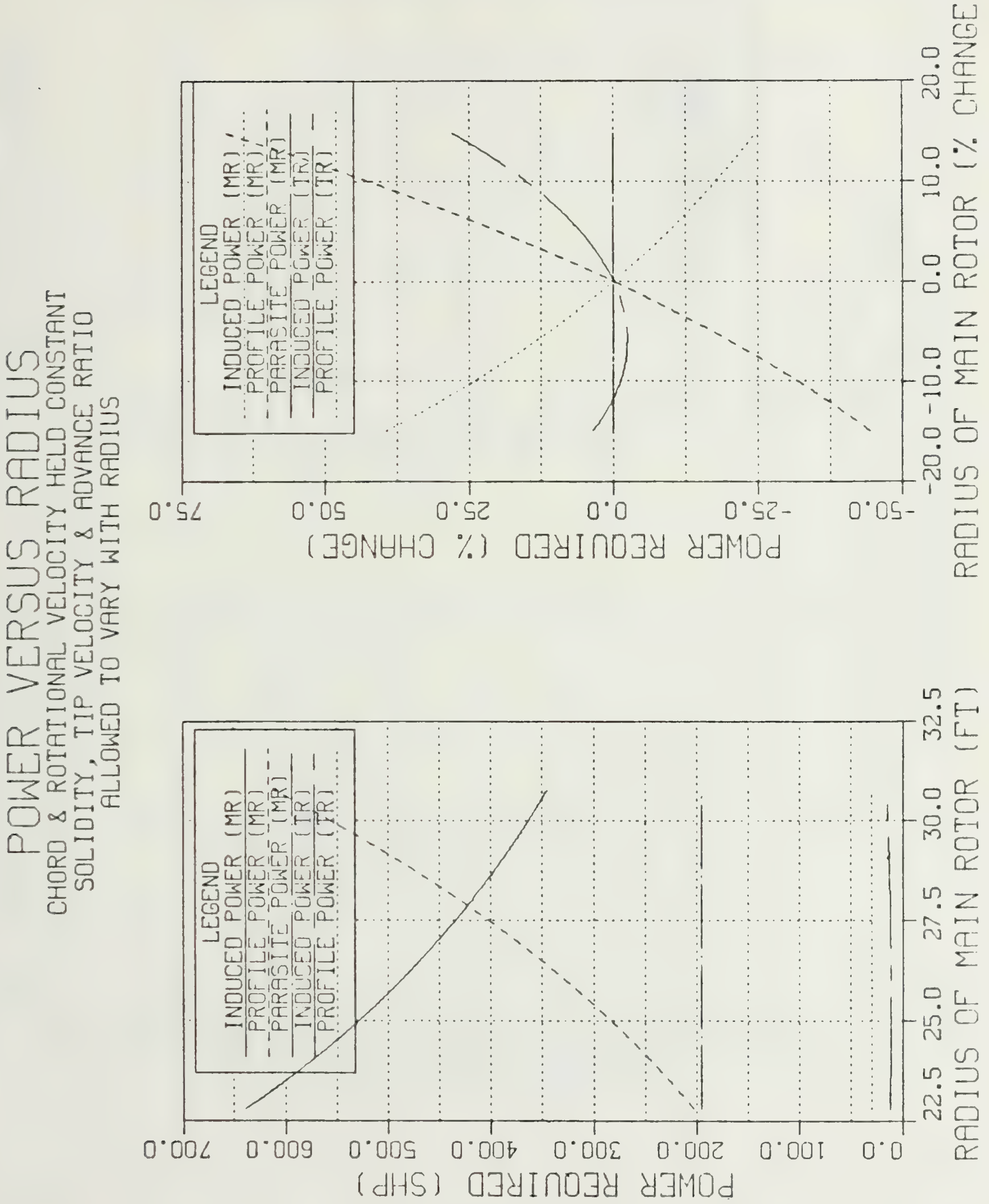
V. CONCLUSIONS

The objective of this thesis to design a program that evaluates a given set of helicopter design values was accomplished. The program will generate thirty two (32) graphs of six (6) performance factors for any single-mast helicopter. The program does allow for one to optimize a design through the evaluation of the variation of parameters for each performance factor.

At this point, the program takes into account neither compressibility nor blade stall. These factors would improve the estimations and may prove to be a worthwhile project for further work. Other points which could be examined are twin-blade and no-tail-rotor helicopters. Of the points mentioned above, blade design would probably be of most importance since little has been done to examine the optimization of blade twist and composite structure of blades.

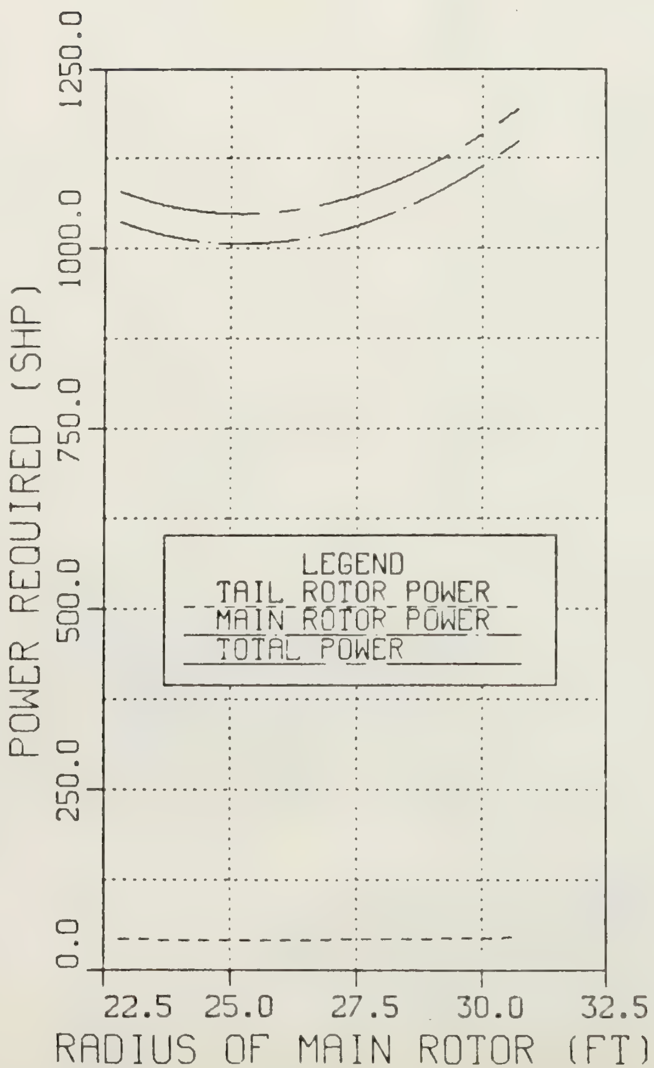
APPENDIX A

COMPUTER PROGRAM OUTPUT - PLOTS



POWER VERSUS RADIUS

CHORD & ROTATIONAL VELOCITY HELD CONSTANT
SOLIDITY, TIP VELOCITY & ADVANCE RATIO
ALLOWED TO VARY WITH RADIUS



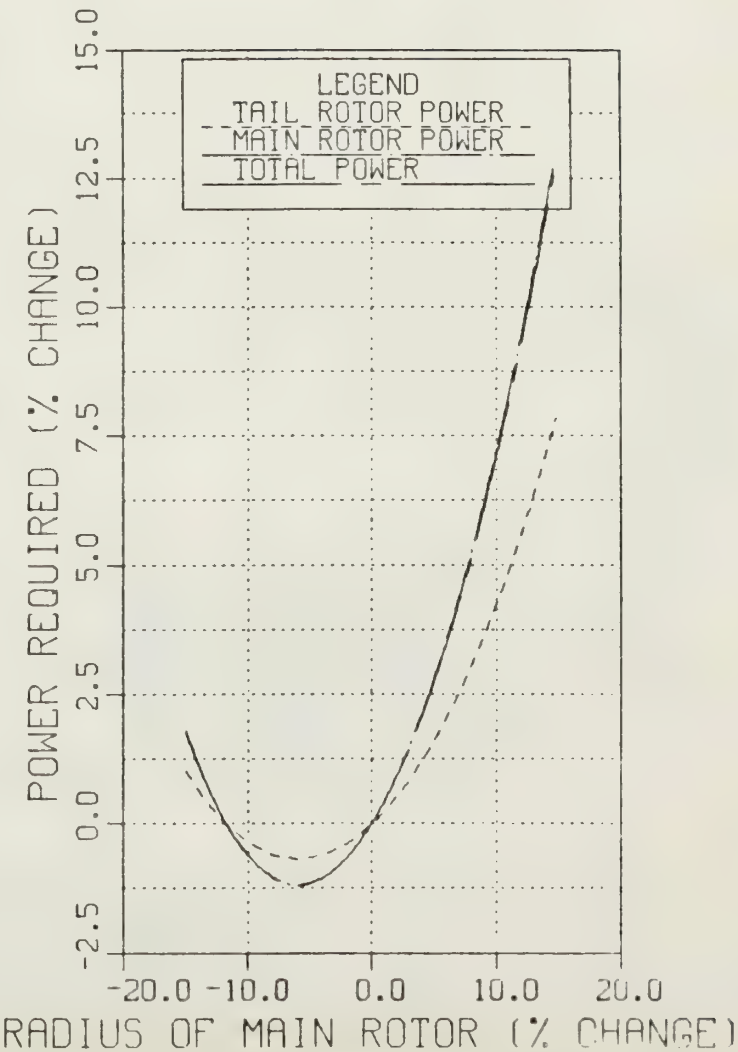
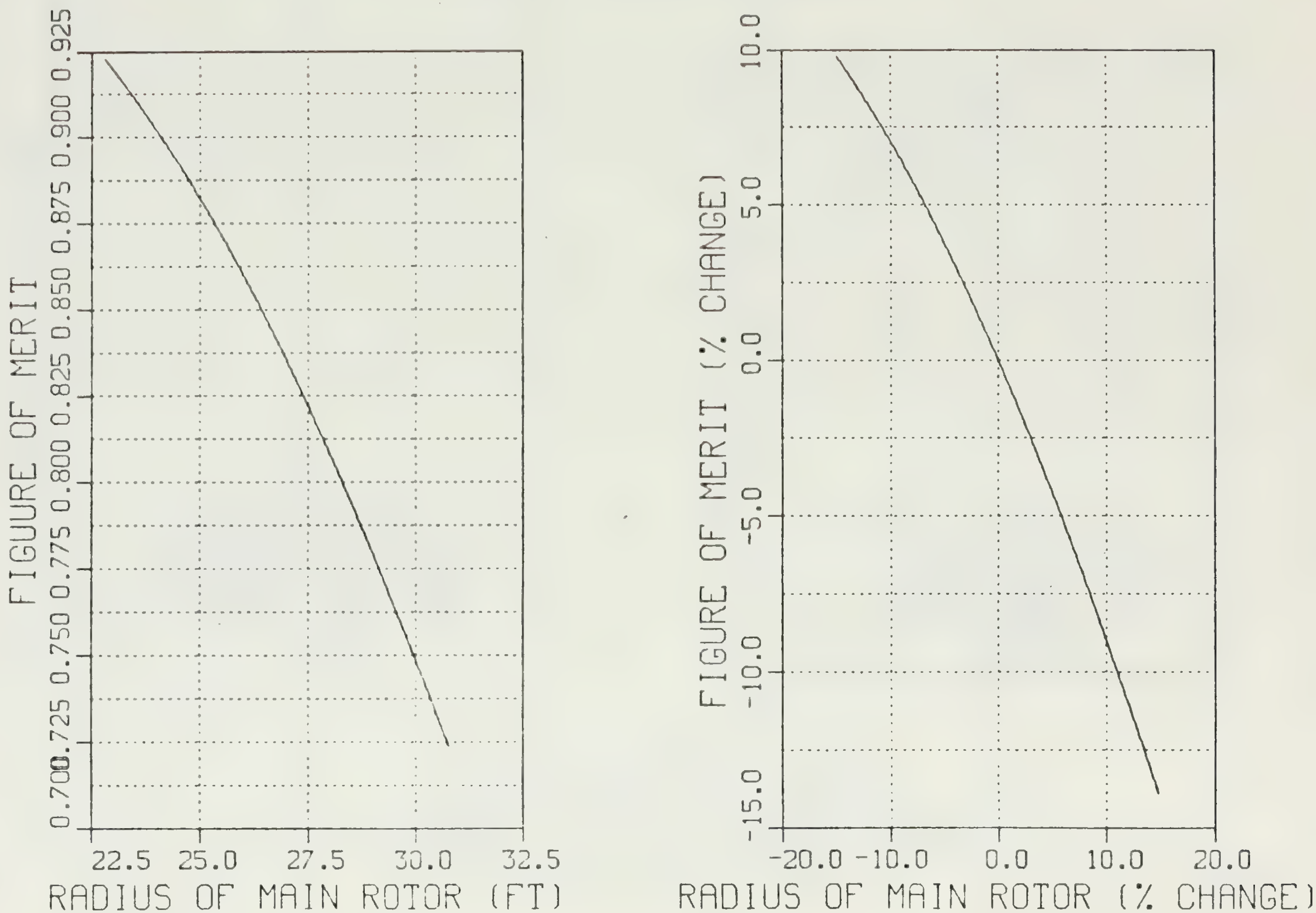


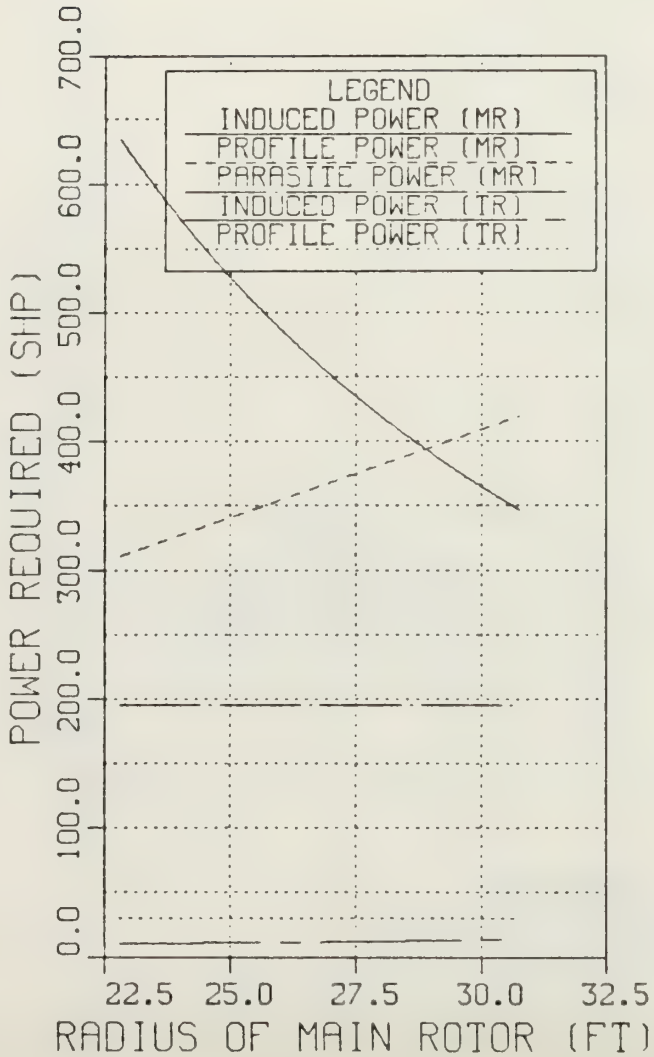
FIGURE OF MERIT VERSUS RADIUS CHANGE

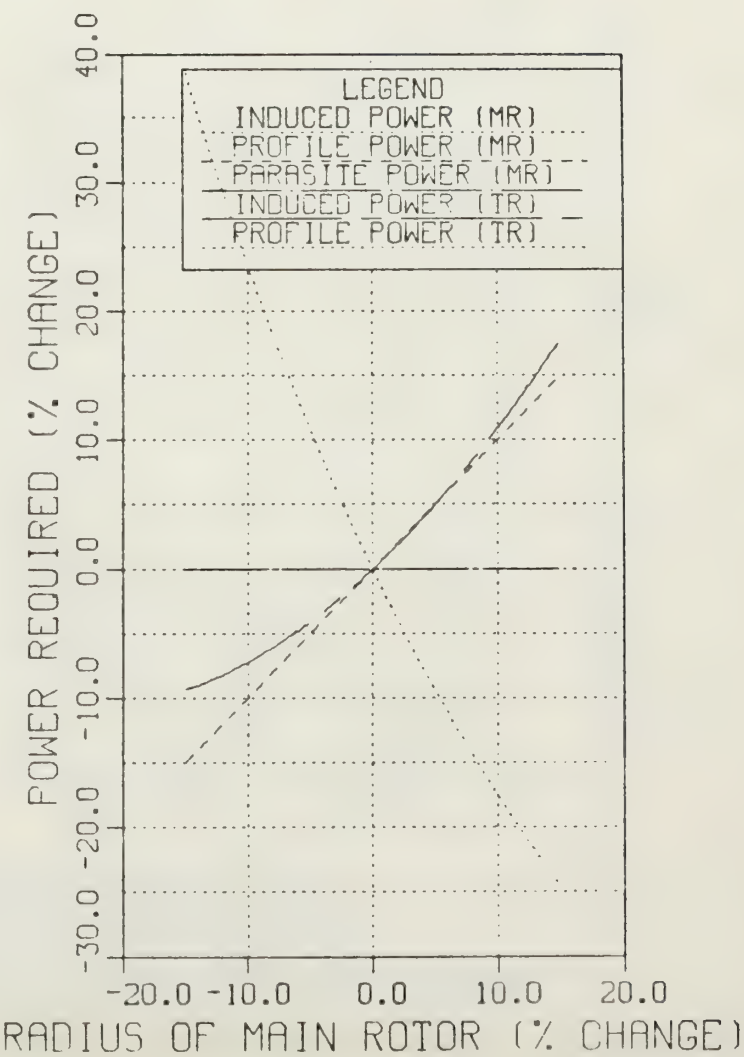
CHORD & ROTATIONAL VELOCITY HELD CONSTANT
SOLIDITY, TIP VELOCITY & ADVANCE RATIO
ALLOWED TO VARY WITH RADIUS



POWER VERSUS RADIUS

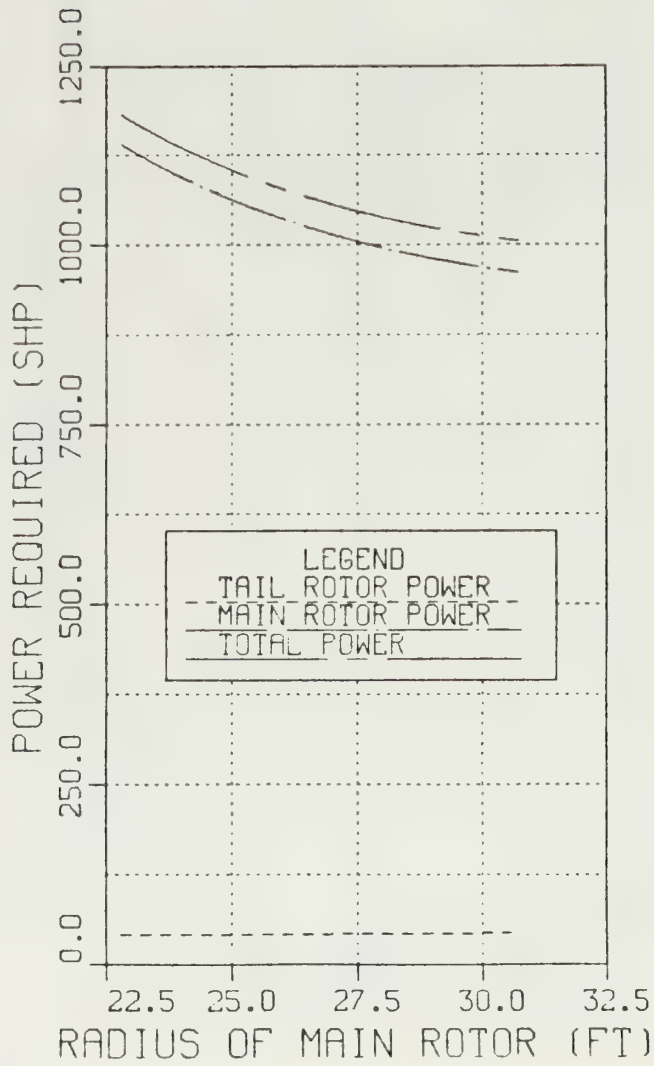
CHORD, TIP VELOCITY & ADVANCE RATIO HELD CONSTANT
SOLIDITY, ROTATIONAL VELOCITY
ALLOWED TO VARY WITH RADIUS





POWER VERSUS RADIUS

CHORD, TIP VELOCITY & ADVANCE RATIO HELD CONSTANT
SOLIDITY, ROTATIONAL VELOCITY
ALLOWED TO VARY WITH RADIUS



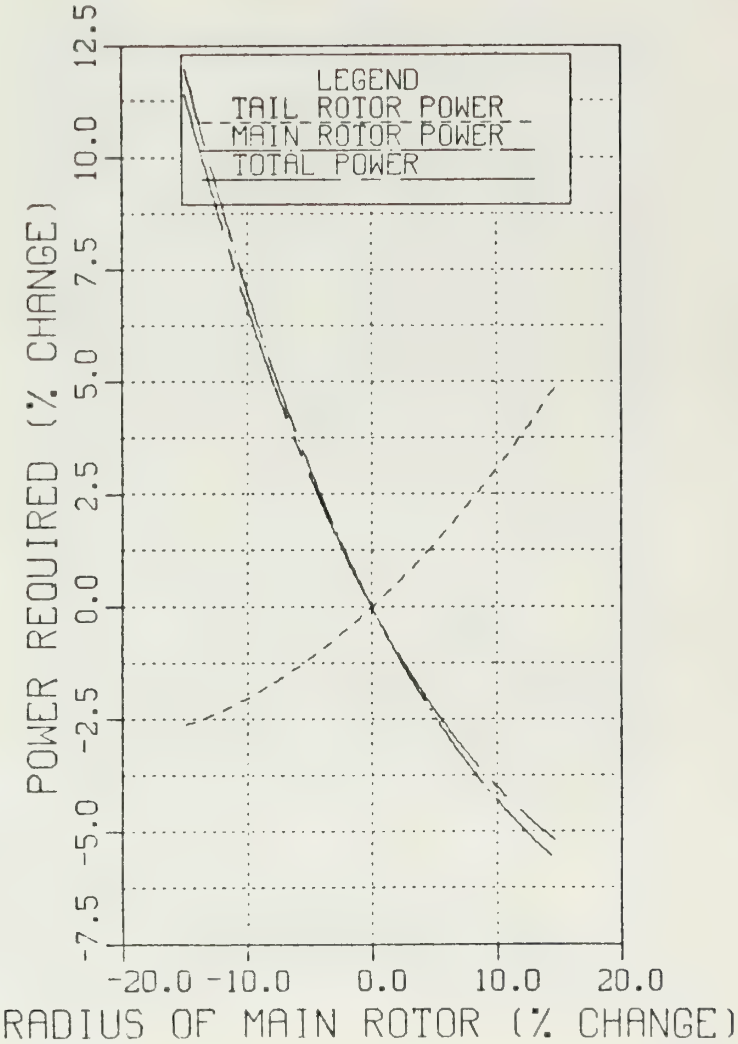
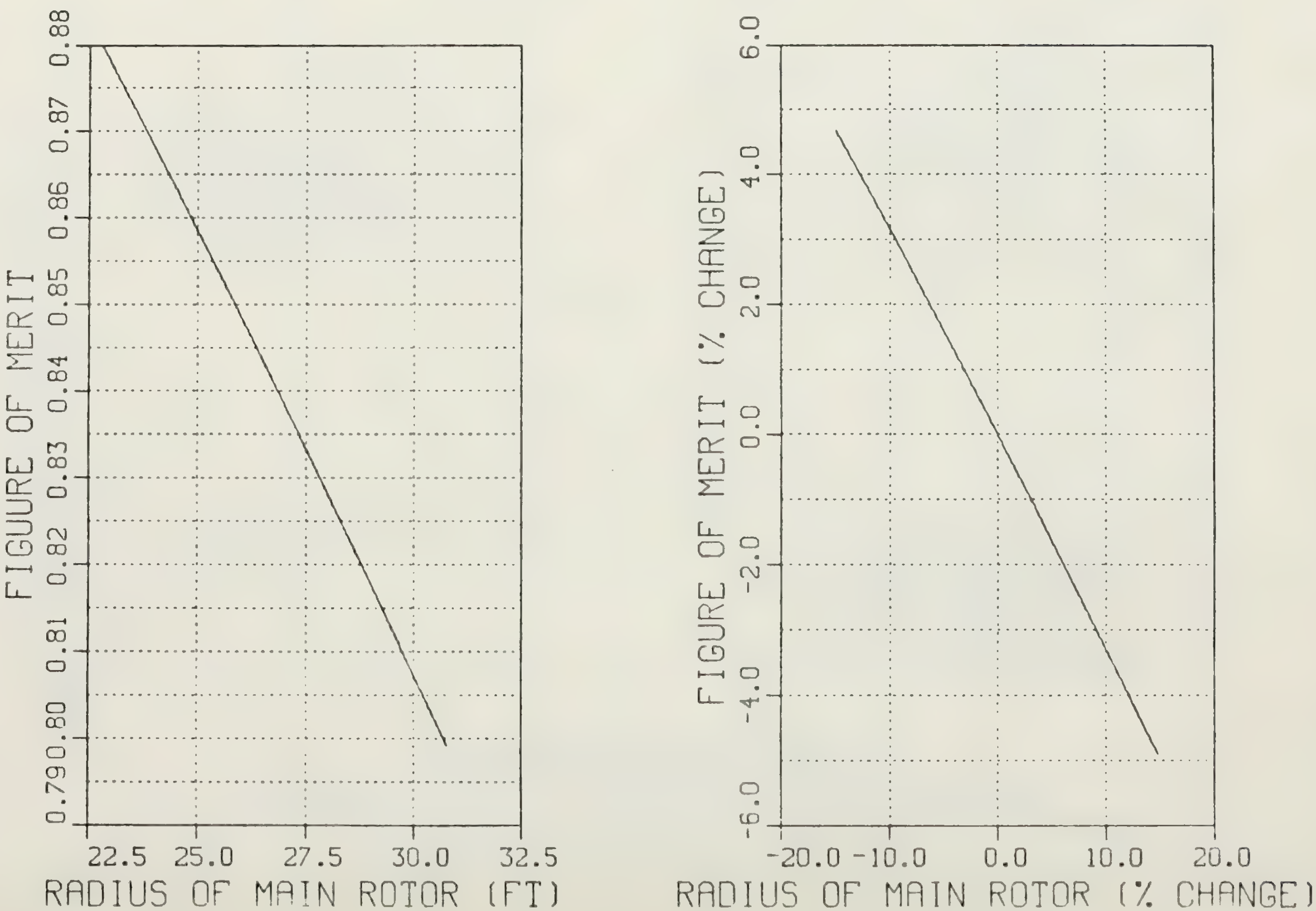
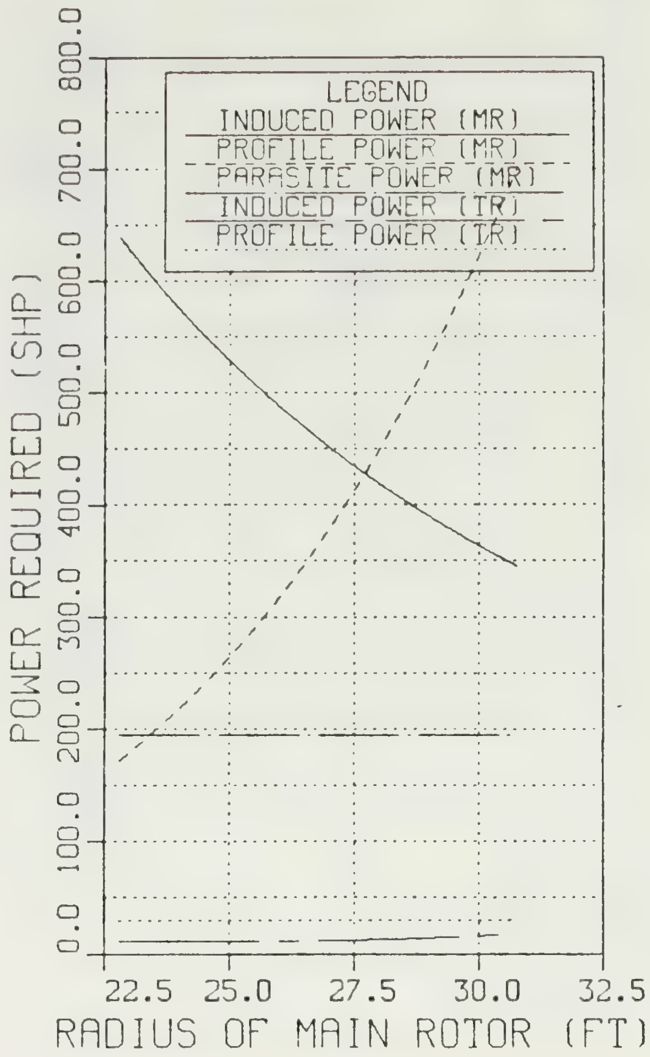


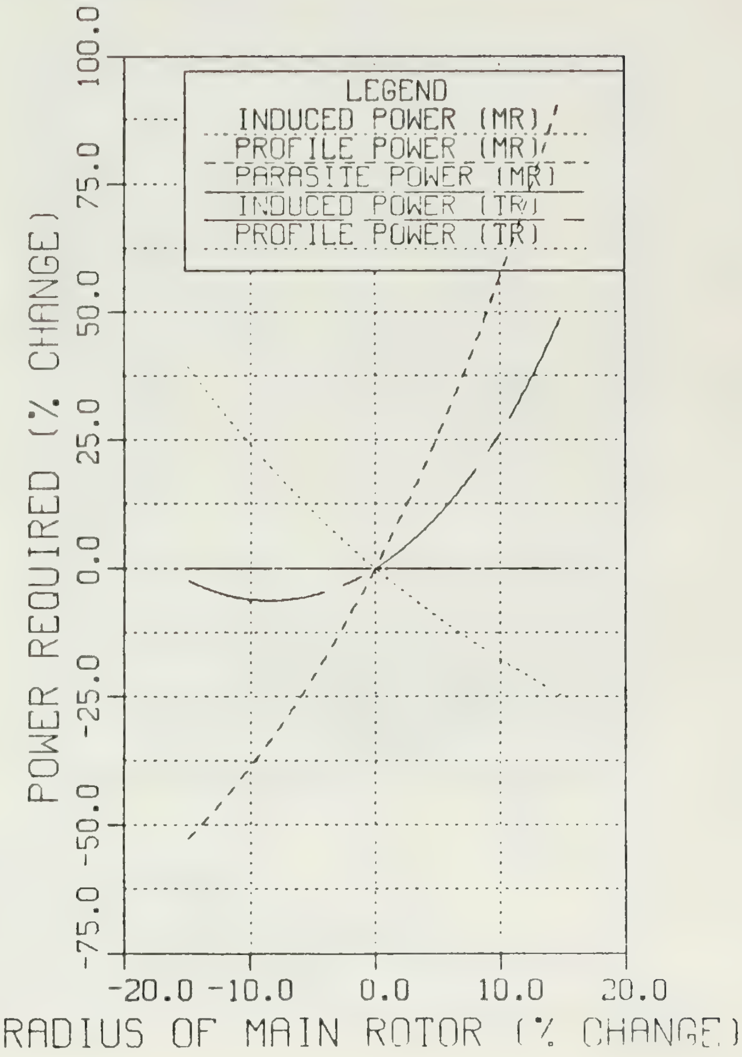
FIGURE OF MERIT VERSUS RADIUS CHANGE
CHORD,TIP VELOCITY & ADVANCE RATIO HELD CONSTANT
SOLIDITY, ROTATIONAL VELOCITY
ALLOWED TO VARY WITH RADIUS



POWER VERSUS RADIUS

SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT
CHORD, ADVANCE RATIO & TIP VELOCITY
ALLOWED TO VARY WITH RADIUS





POWER VERSUS RADIUS

SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT
CHORD,ADVANCE RATIO & TIP VELOCITY
ALLOWED TO VARY WITH RADIUS

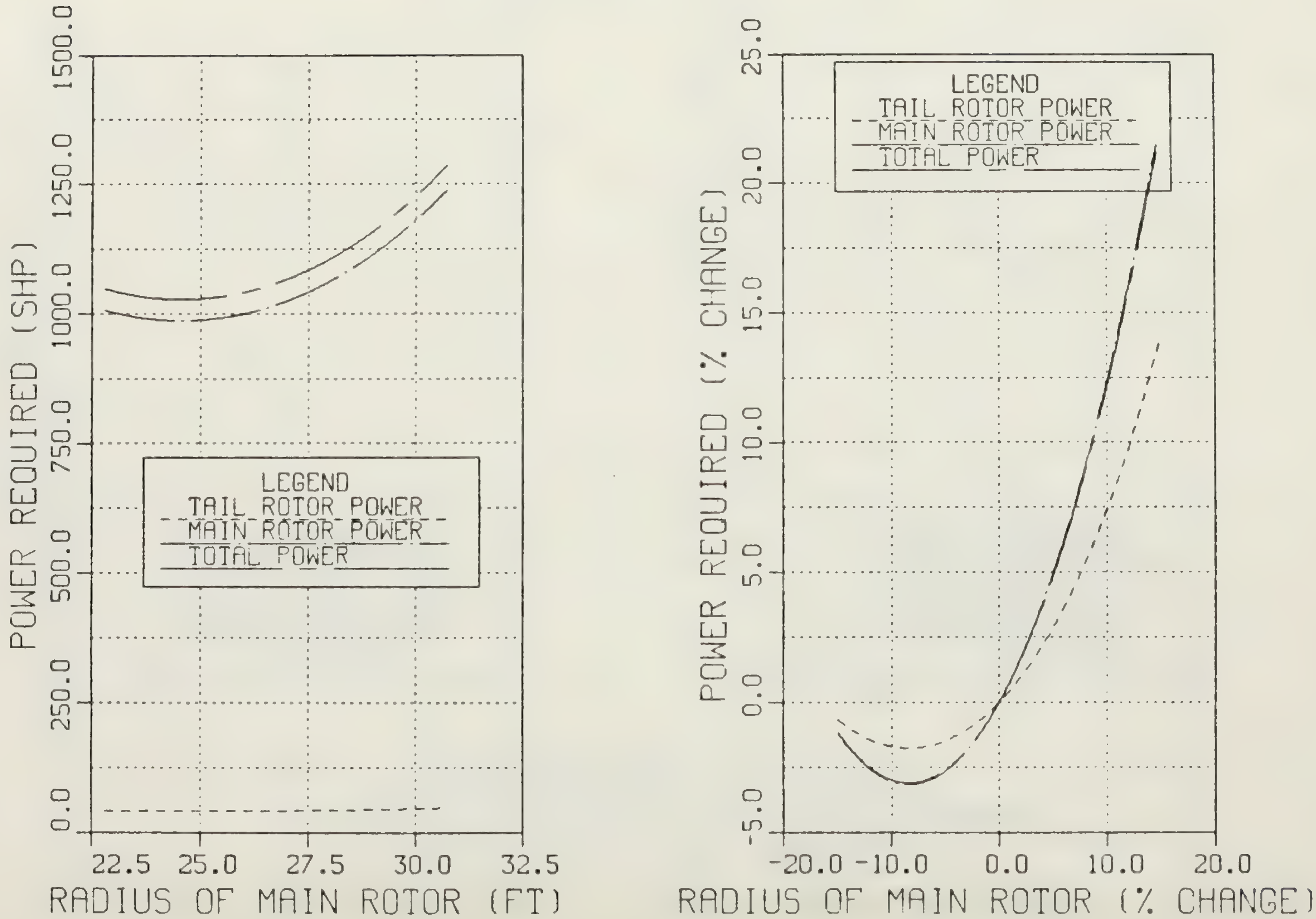
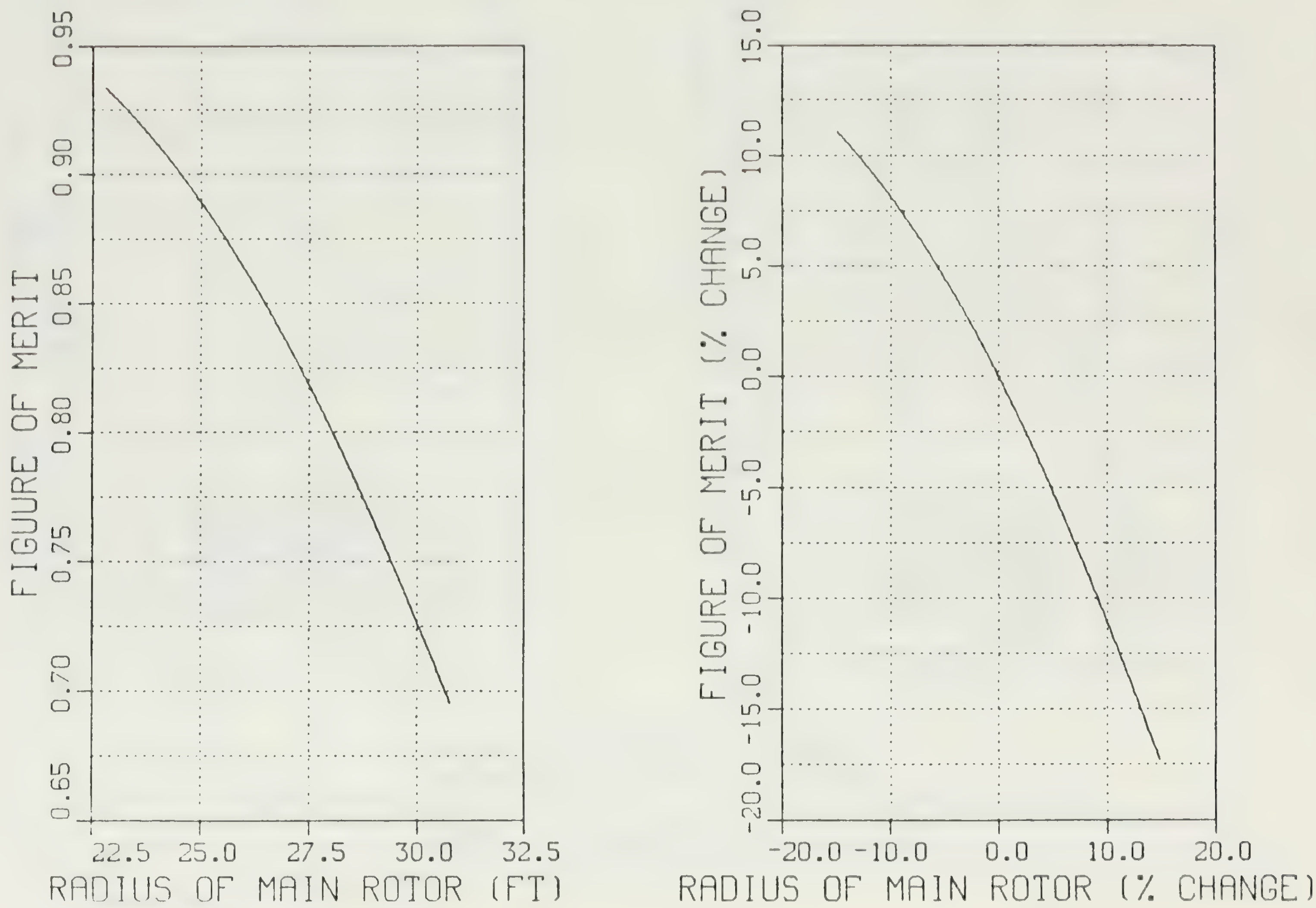
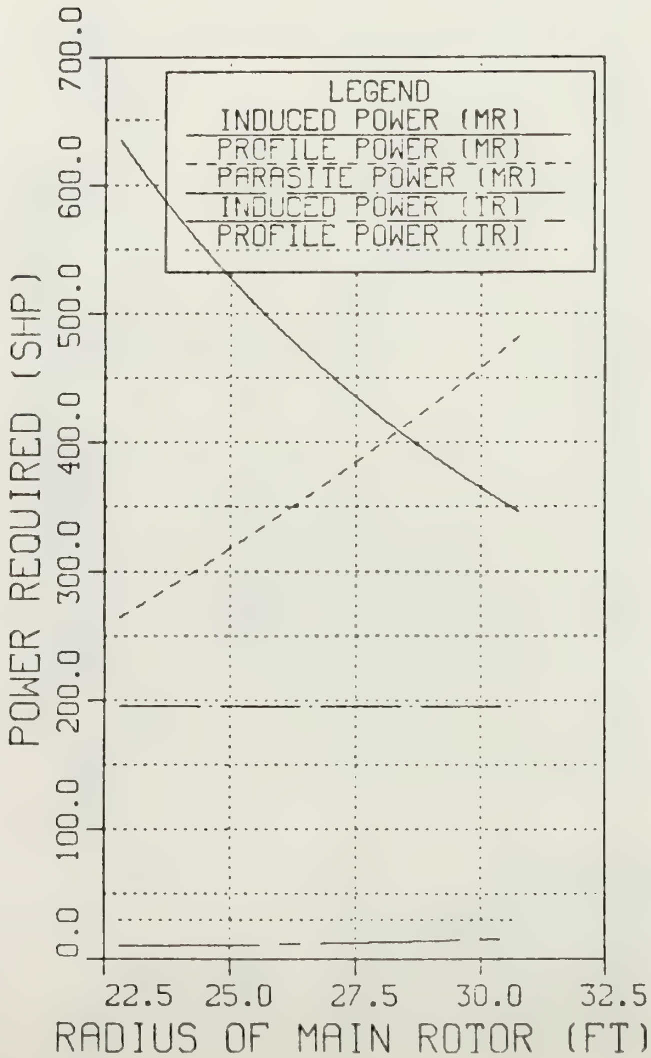


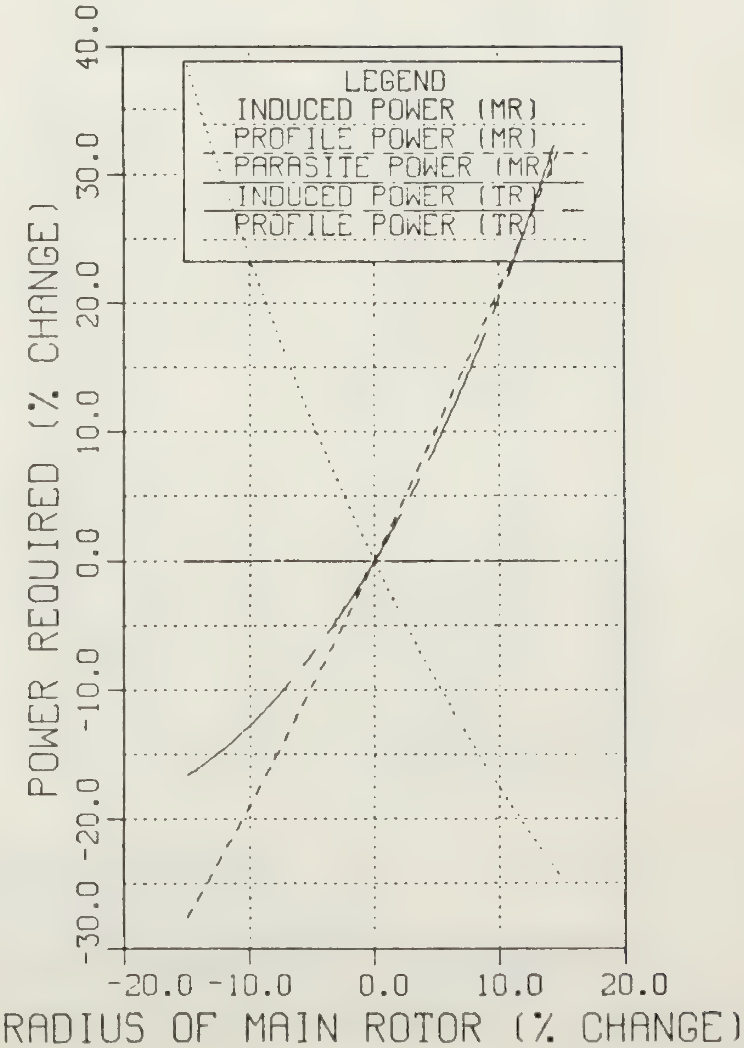
FIGURE OF MERIT VERSUS RADIUS CHANGE
SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT
CHORD,ADVANCE RATIO & TIP VELOCITY
ALLOWED TO VARY WITH RADIUS



POWER VERSUS RADIUS

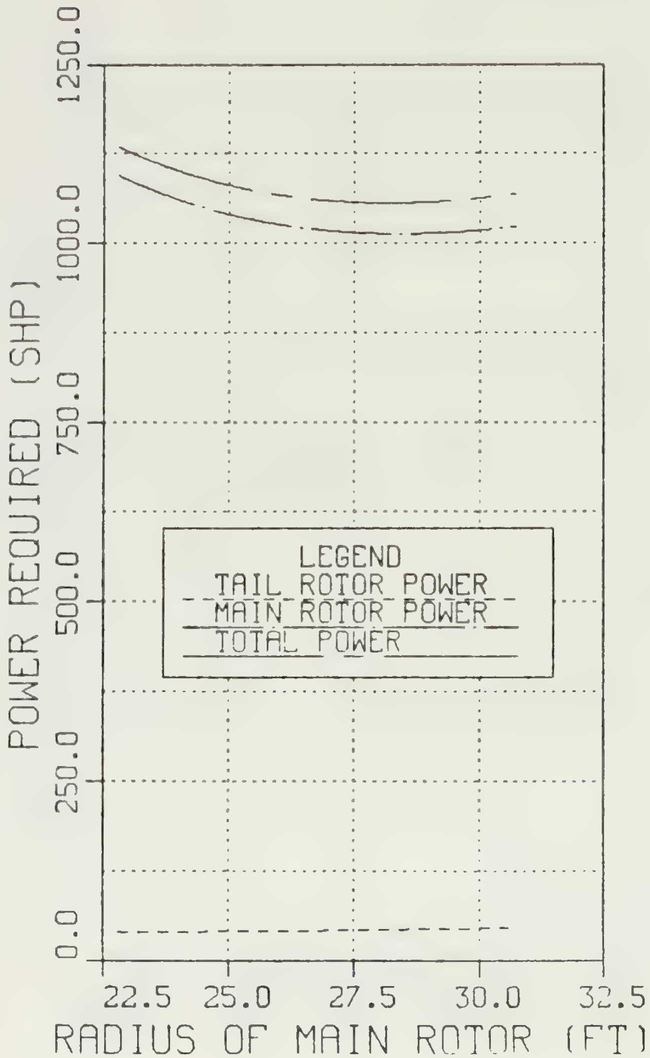
SOLIDITY, ADVANCE RATIO & TIP VELOCITY HELD CONSTANT
CHORD & ROTATIONAL VELOCITY
ALLOWED TO VARY WITH RADIUS





POWER VERSUS RADIUS

SOLIDITY, ADVANCE RATIO & TIP VELOCITY HELD CONSTANT
CHORD & ROTATIONAL VELOCITY
ALLOWED TO VARY WITH RADIUS



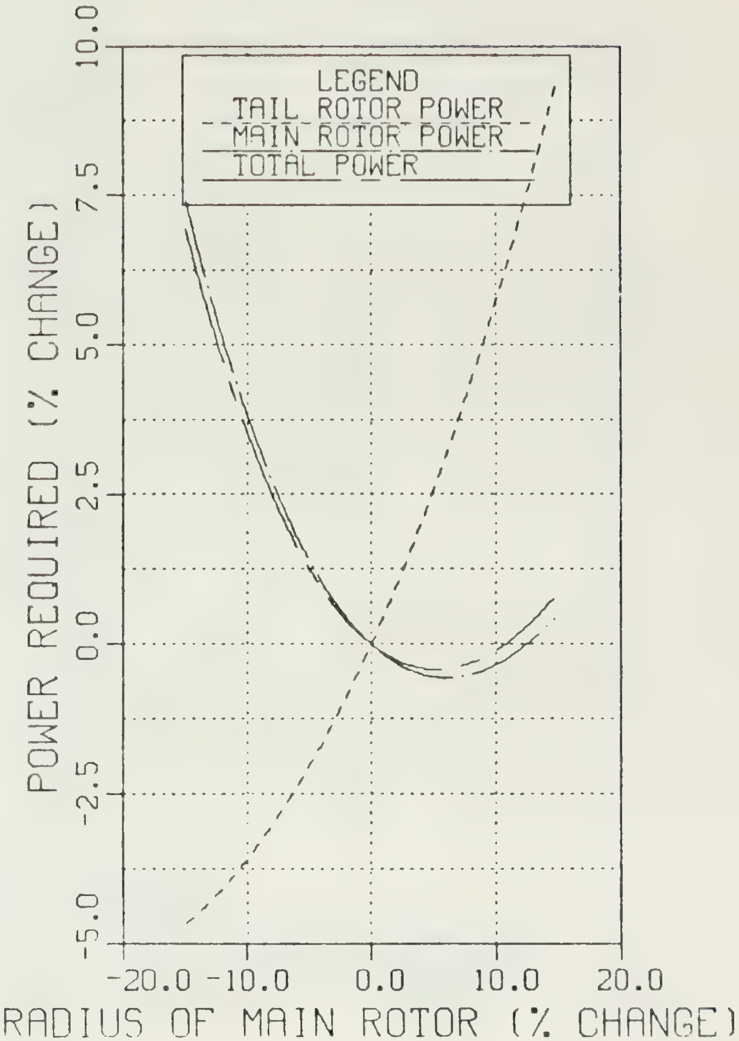
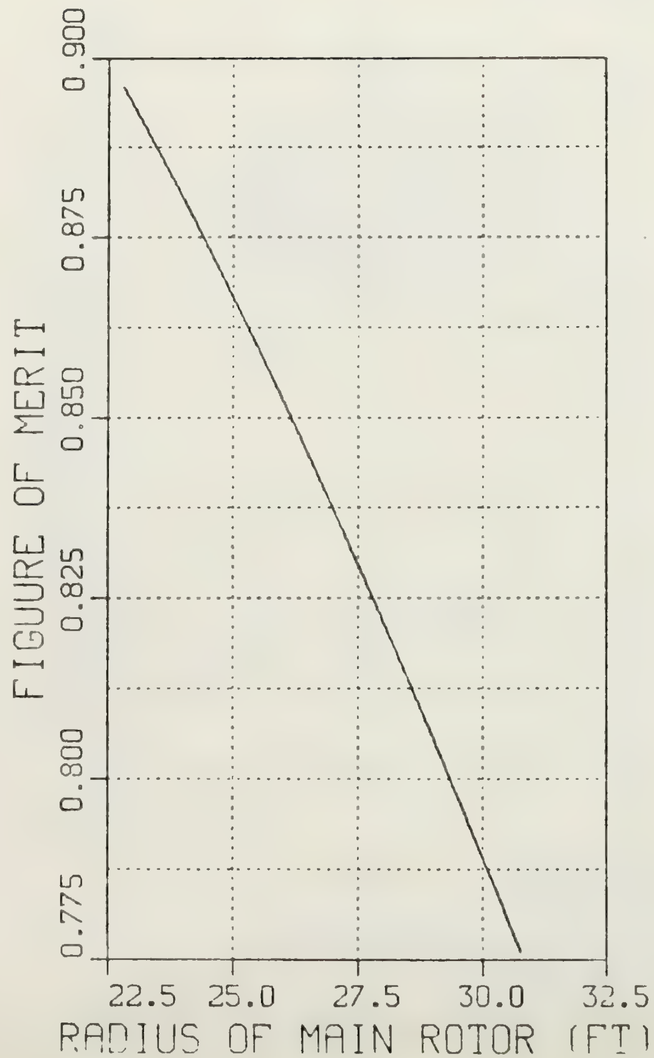
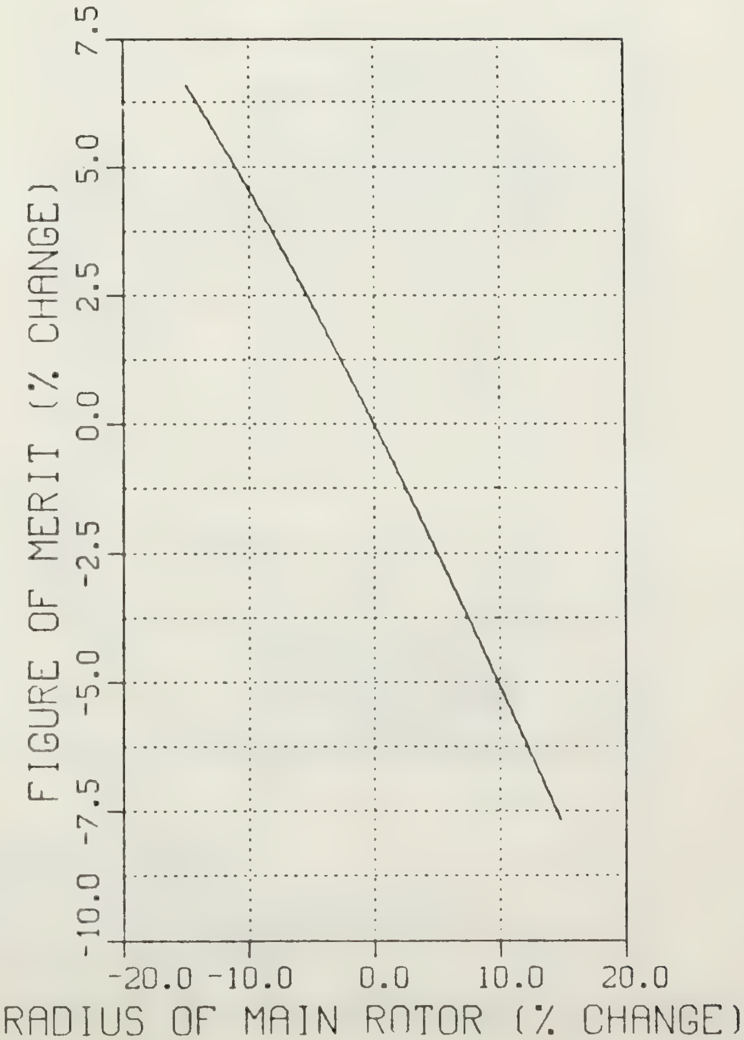


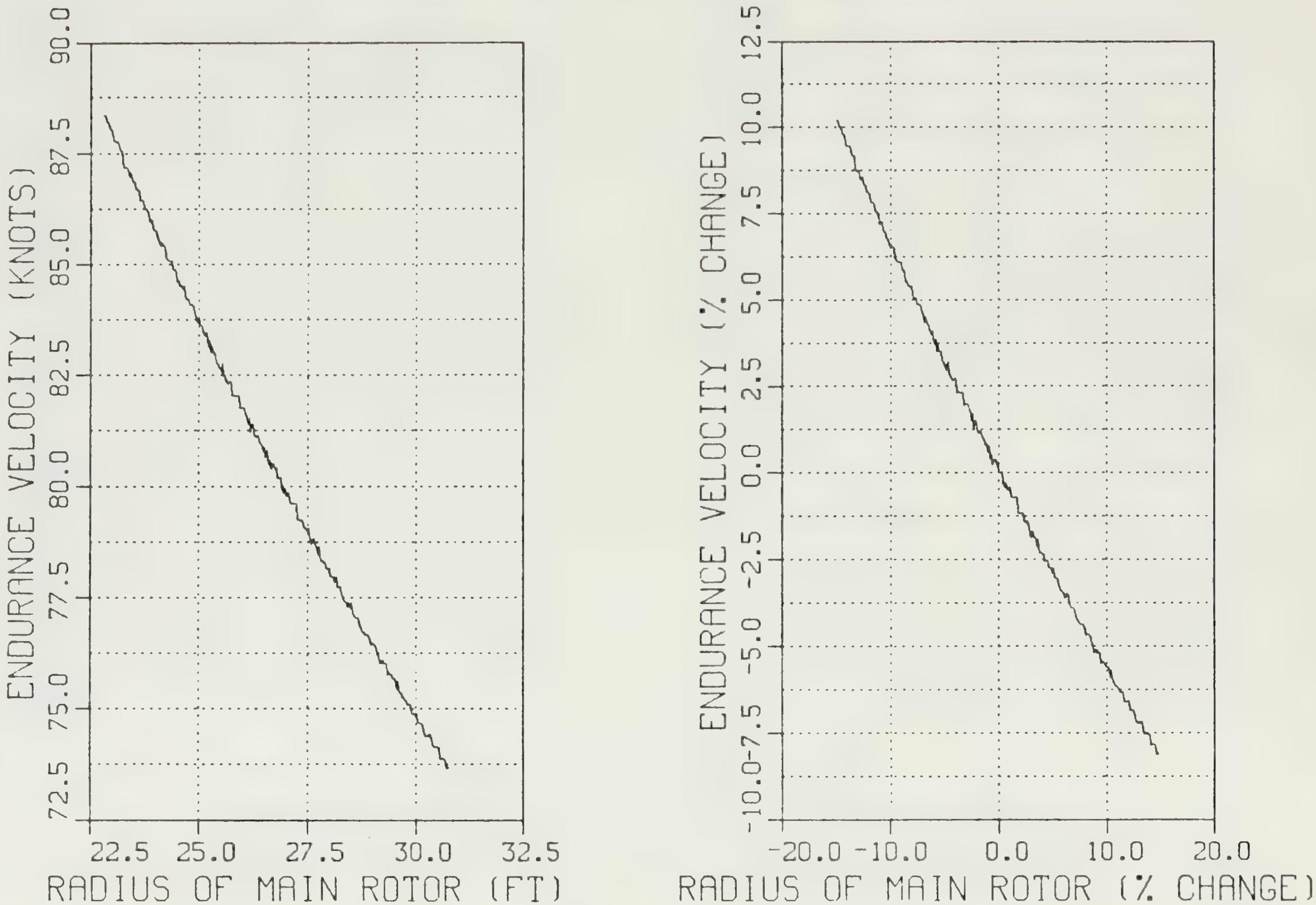
FIGURE OF MERIT VERSUS RADIUS CHANGE

SOLIDITY, ADVANCE RATIO & TIP VELOCITY HELD CONSTANT
CHORD & ROTATIONAL VELOCITY
ALLOWED TO VARY WITH RADIUS

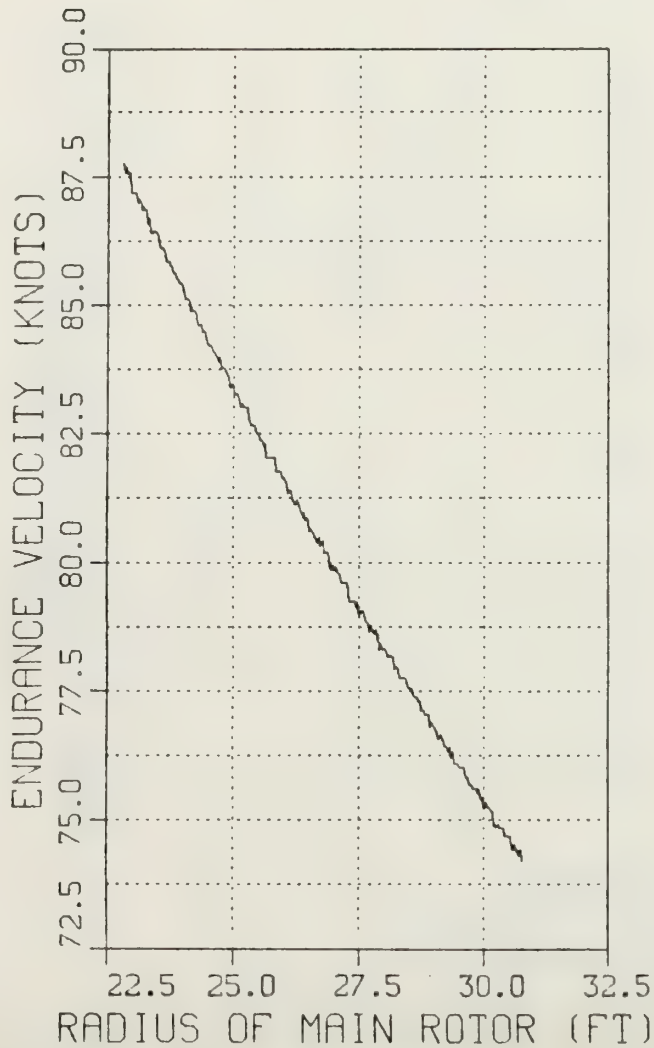


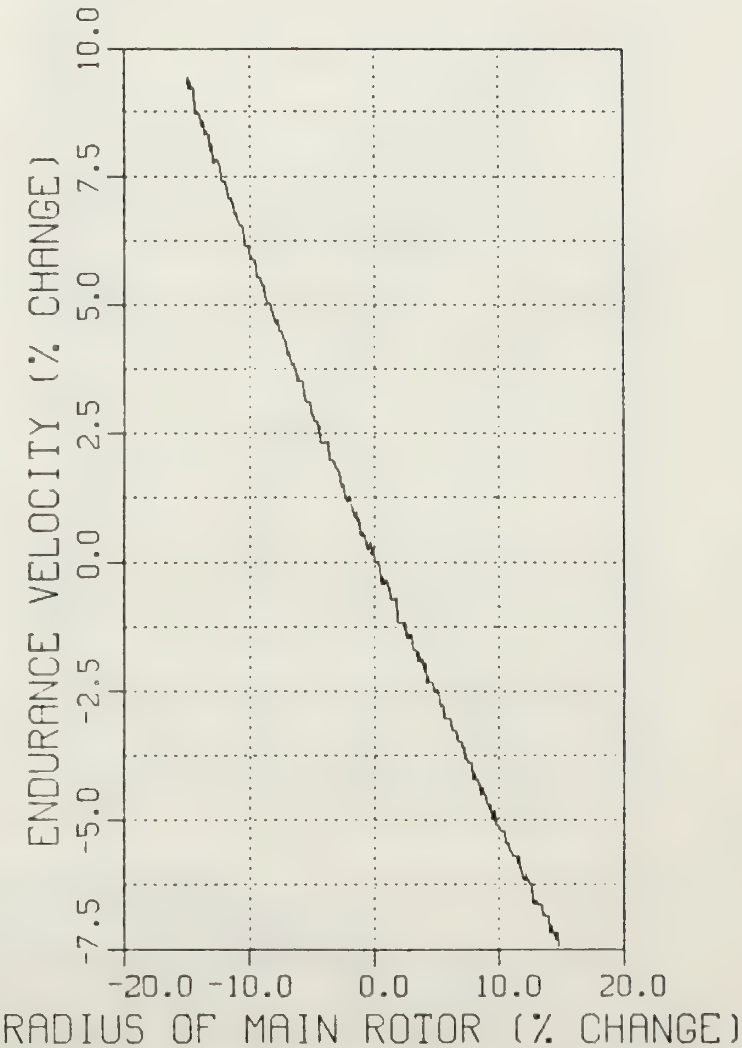


ENDURANCE VELOCITY VERSUS RADIUS
CHORD & ROTATIONAL VELOCITY HELD CONSTANT
SOLIDITY, TIP VELOCITY & ADVANCE RATIO
ALLOWED TO VARY WITH RADIUS



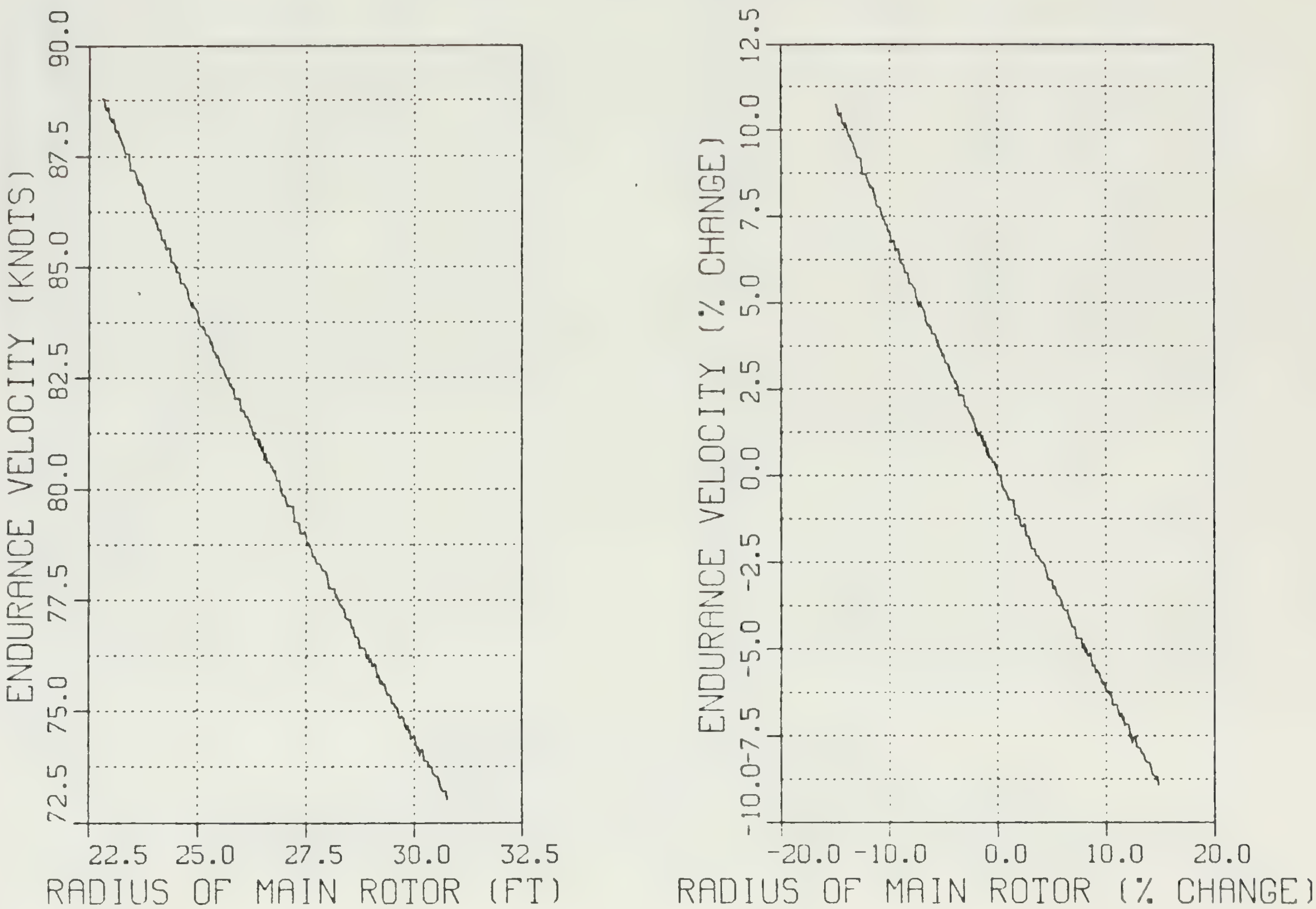
ENDURANCE VELOCITY VERSUS RADIUS
CHORD,TIP VELOCITY & ADVANCE RATIO HELD CONSTANT
SOLIDITY, ROTATIONAL VELOCITY
ALLOWED TO VARY WITH RADIUS



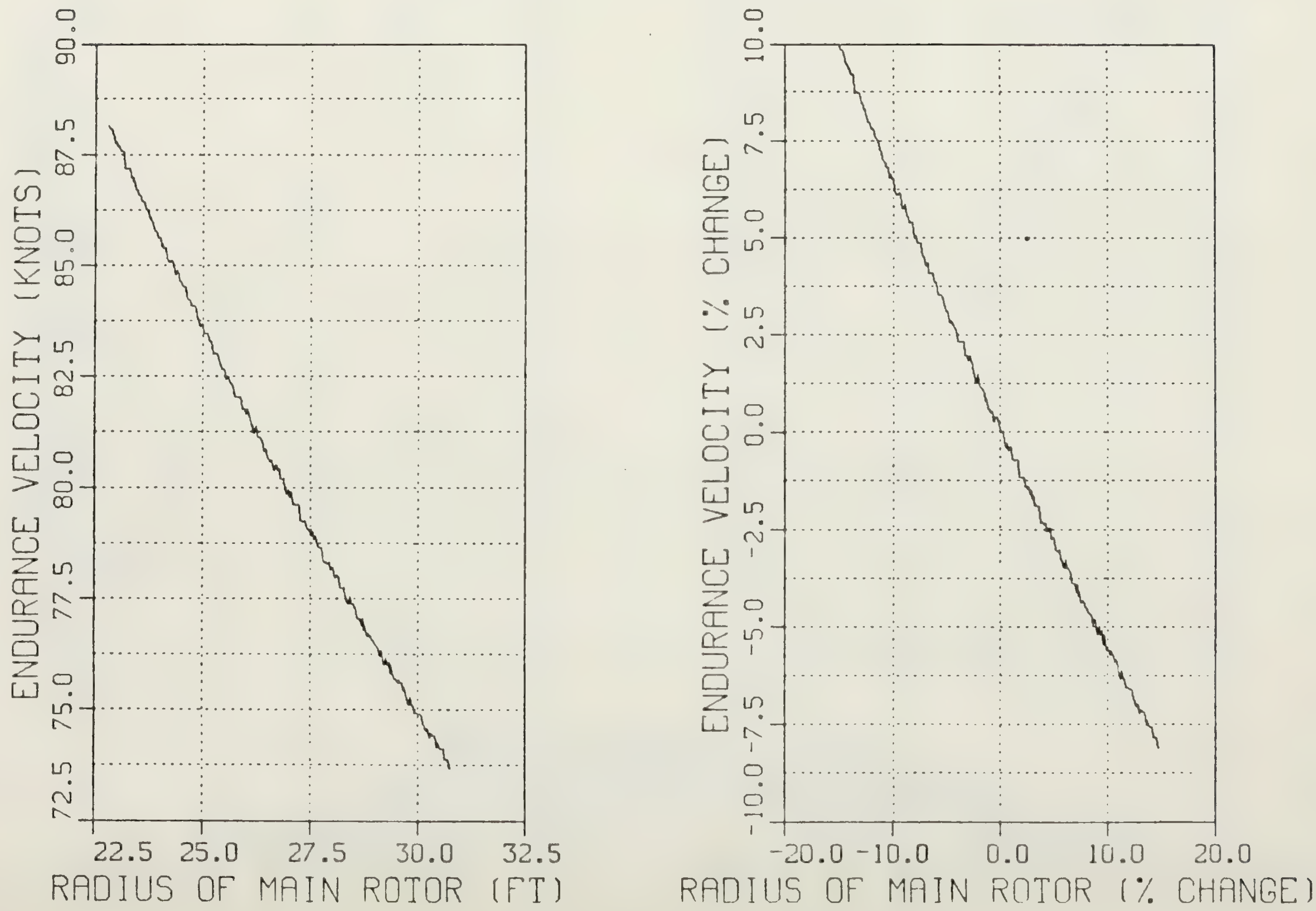


ENDURANCE VELOCITY VERSUS RADIUS

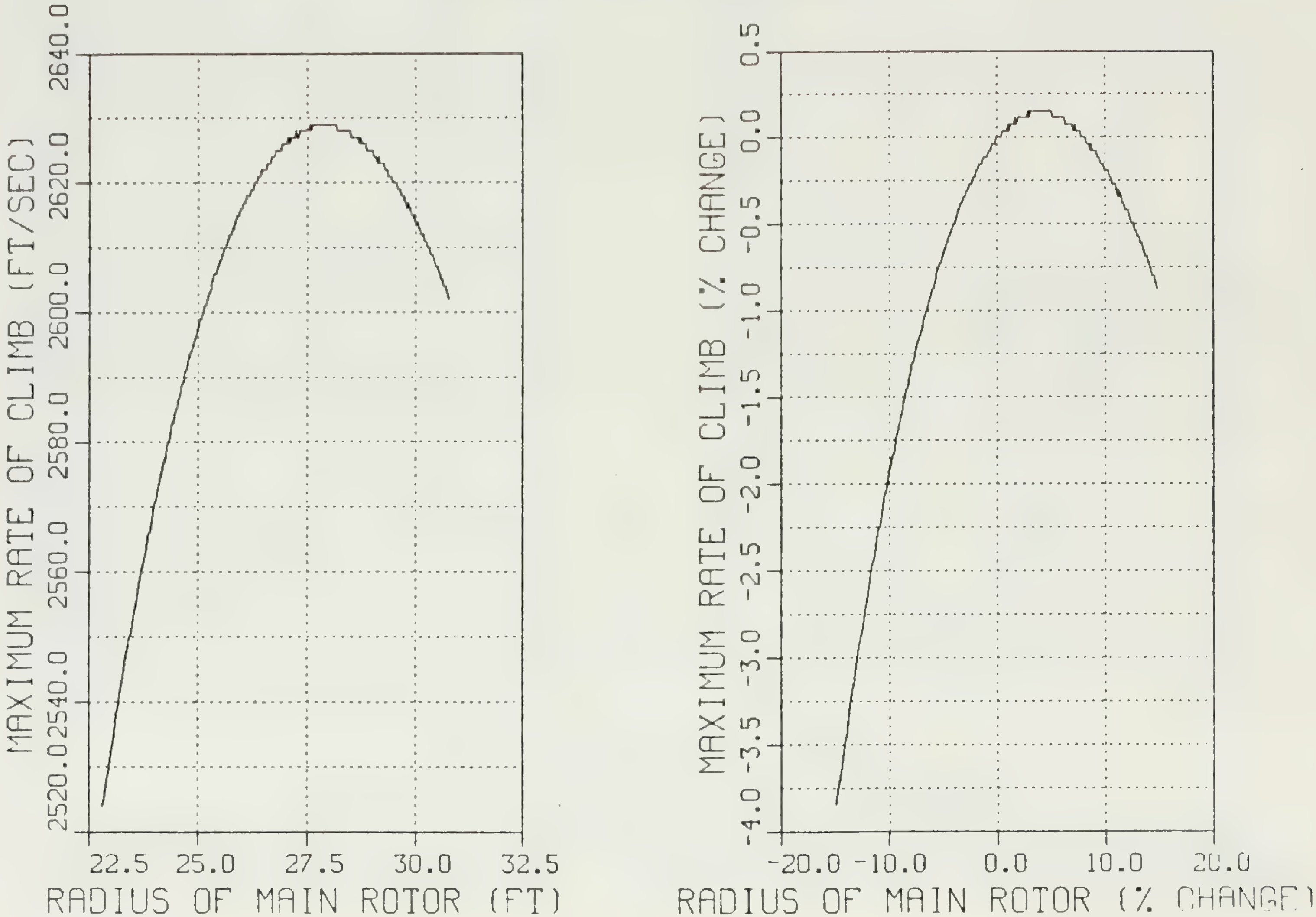
SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT
CHORD, ADVANCE RATIO & TIP VELOCITY
ALLOWED TO VARY WITH RADIUS



ENDURANCE VELOCITY VERSUS RADIUS
SOLIDITY, ADVANCE RATIO & TIP VELOCITY HELD CONSTANT
CHORD & ROTATIONAL VELOCITY
ALLOWED TO VARY WITH RADIUS

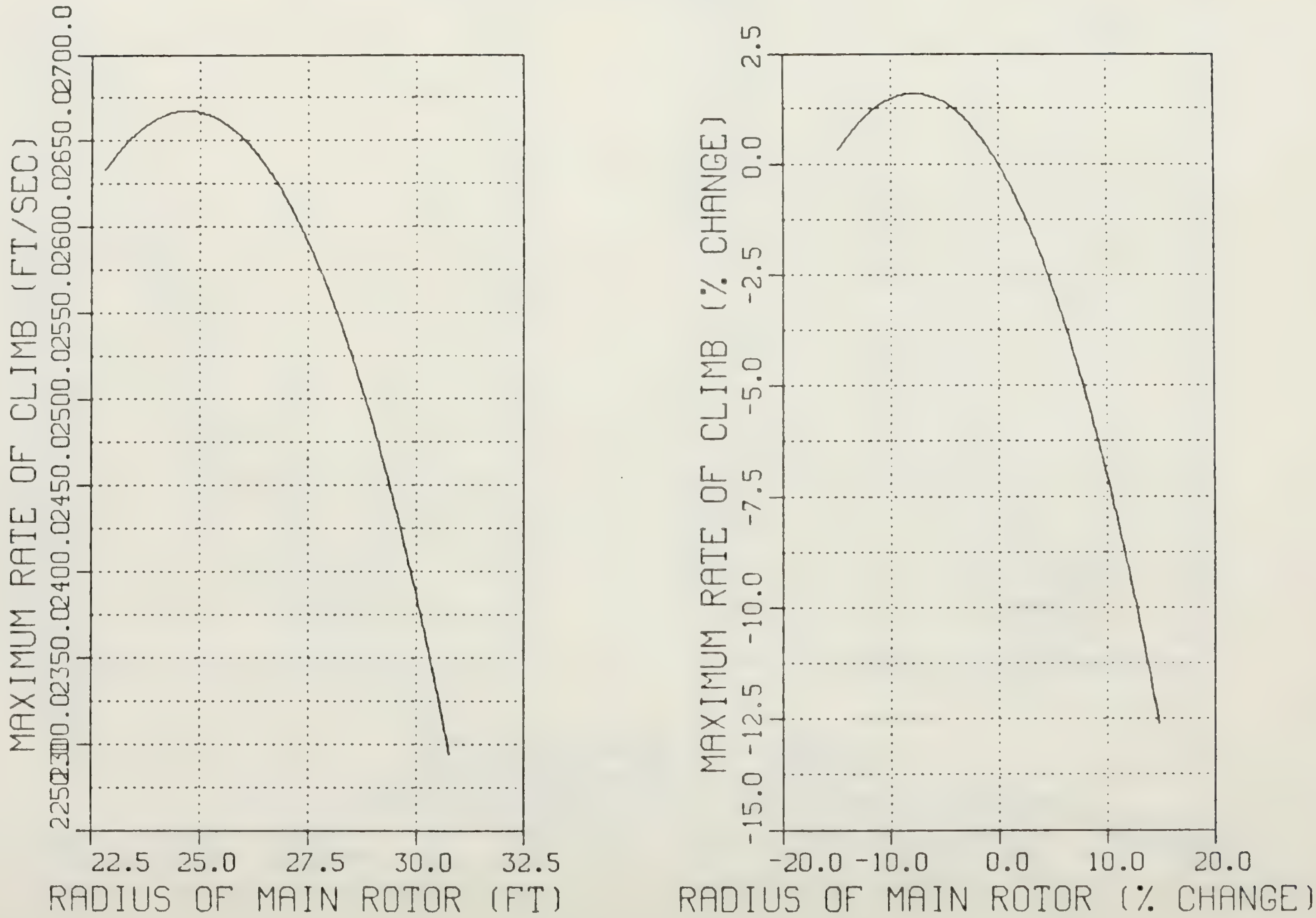


MAXIMUM RATE OF CLIMB VERSUS RADIUS CHANGE
SOLIDITY, ADVANCE RATIO & TIP VELOCITY HELD CONSTANT
CHORD & ROTATIONAL VELOCITY
ALLOWED TO VARY WITH RADIUS



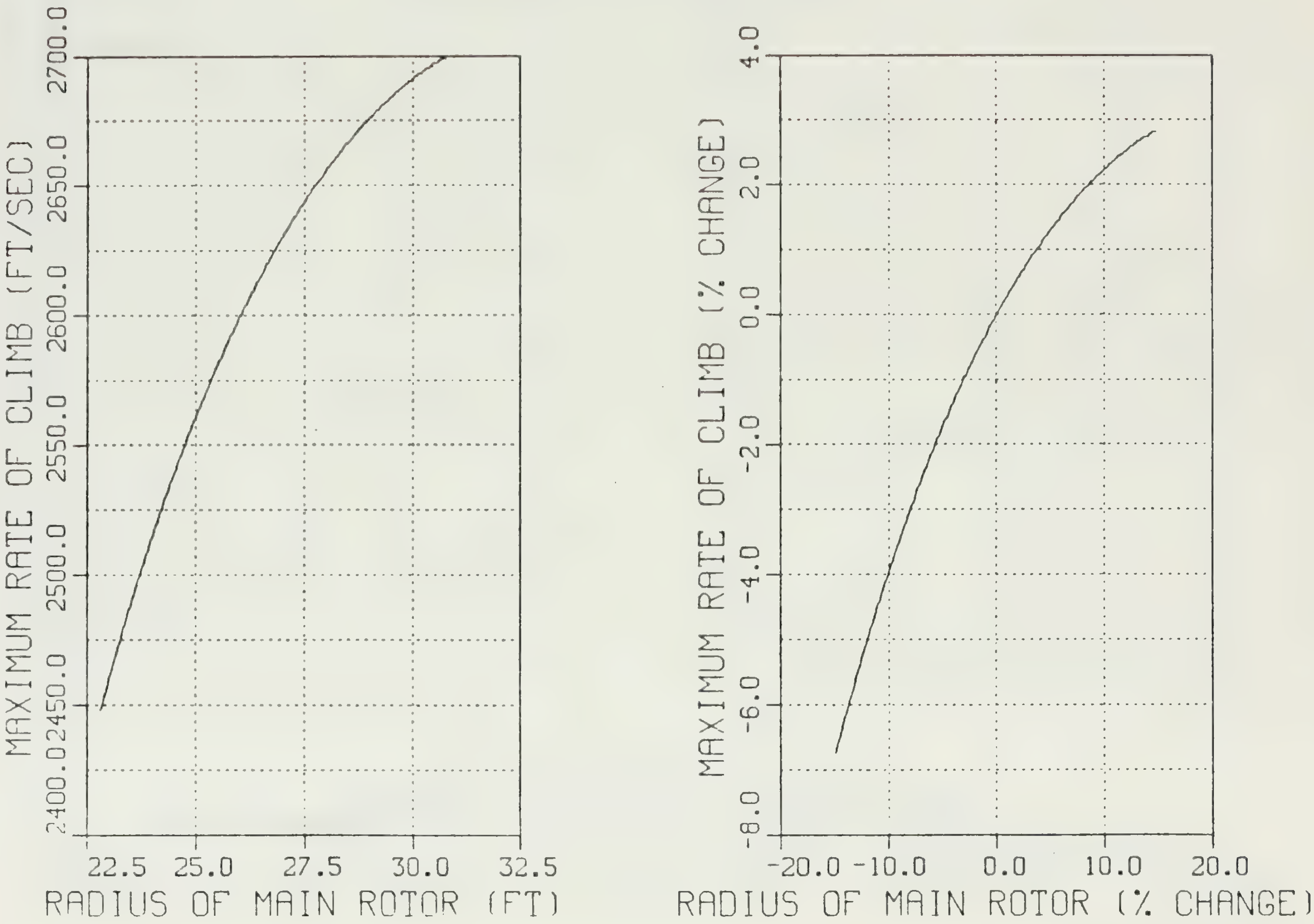
MAXIMUM RATE OF CLIMB VERSUS RADIUS CHANGE

SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT
CHORD, ADVANCE RATIO & TIP VELOCITY
ALLOWED TO VARY WITH RADIUS



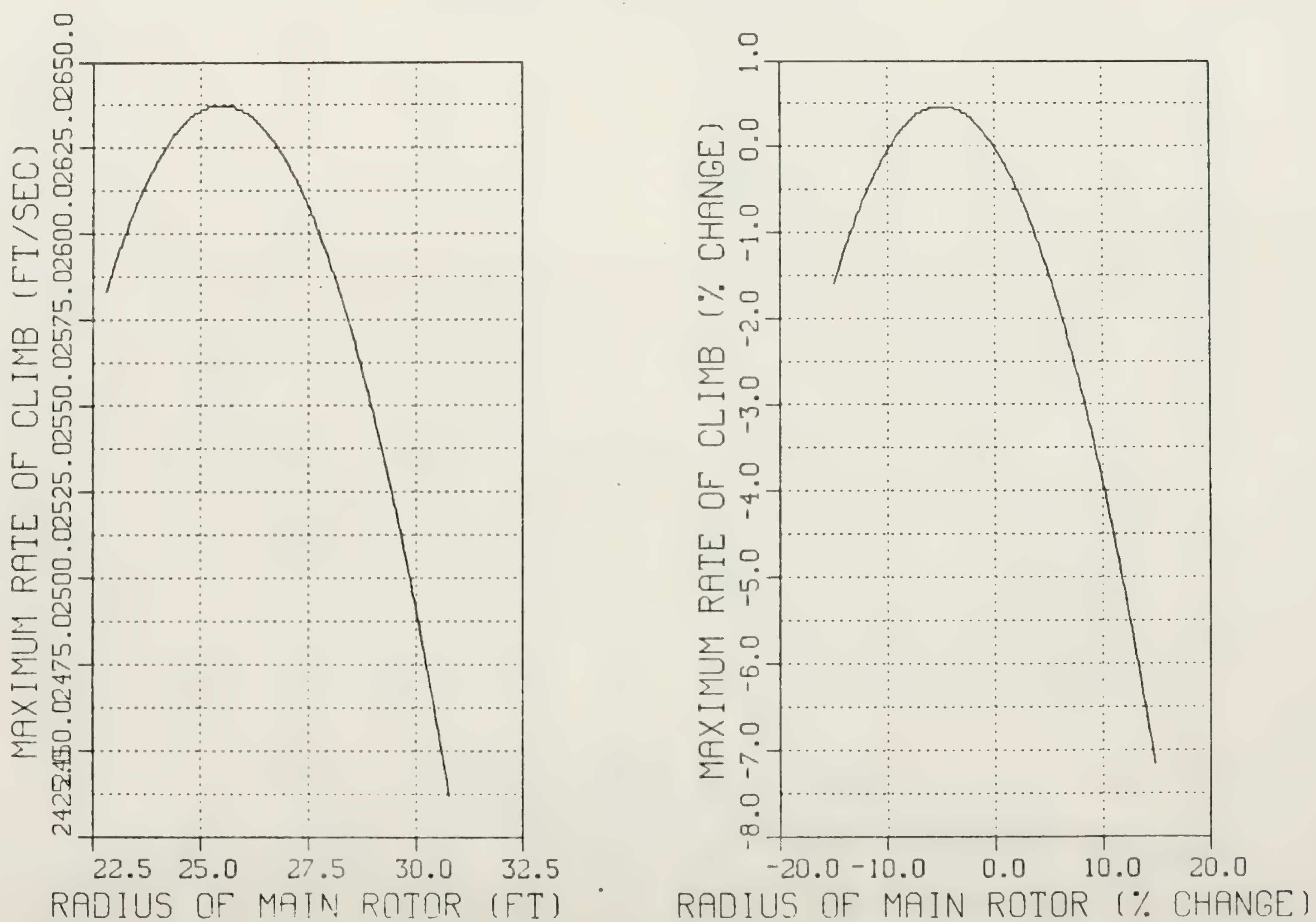
MAXIMUM RATE OF CLIMB VERSUS RADIUS CHANGE

CHORD,TIP VELOCITY & ADVANCE RATIO HELD CONSTANT
SOLIDITY, ROTATIONAL VELOCITY
ALLOWED TO VARY WITH RADIUS



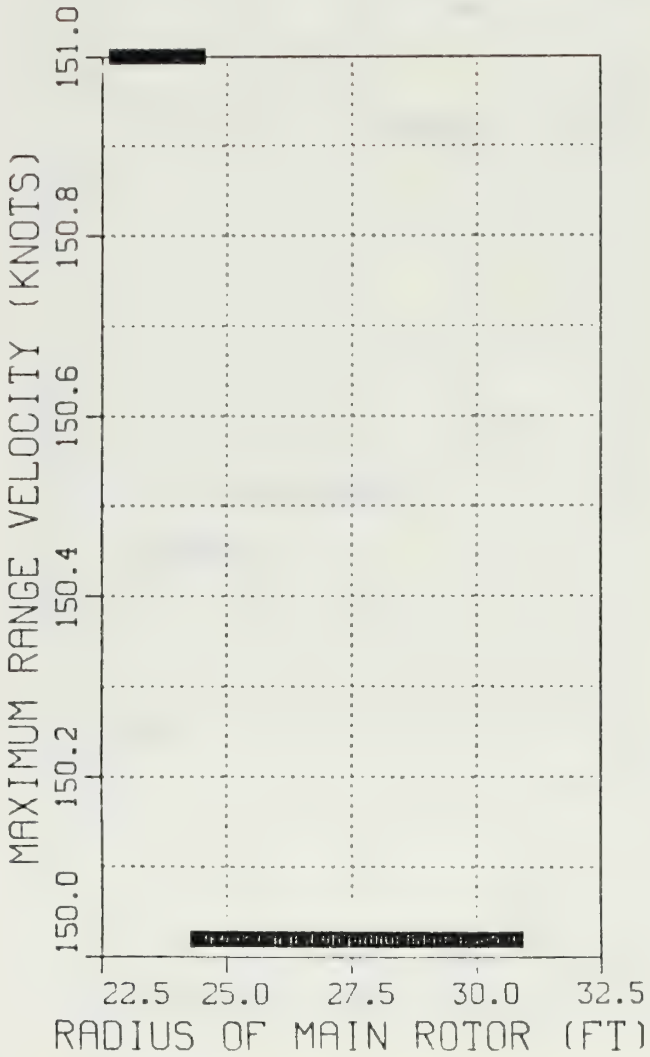
MAXIMUM RATE OF CLIMB VERSUS RADIUS CHANGE

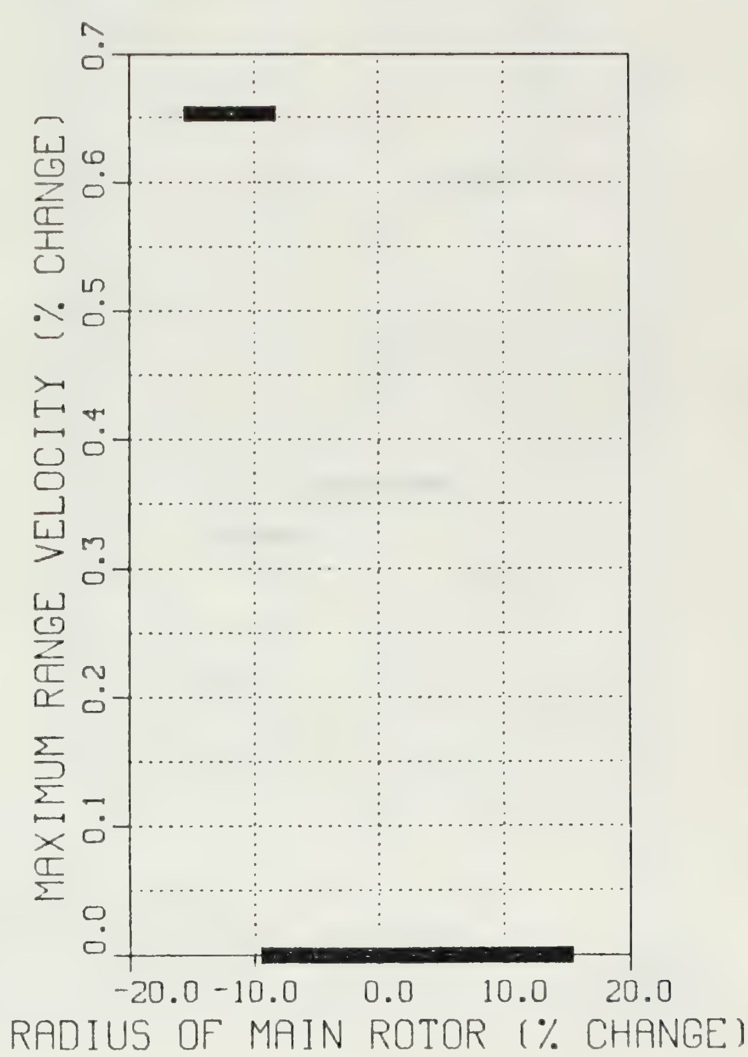
CHORD & ROTATIONAL VELOCITY HELD CONSTANT
SOLIDITY, TIP VELOCITY & ADVANCE RATIO
ALLOWED TO VARY WITH RADIUS



MAXIMUM RANGE VELOCITY VERSUS RADIUS CHANGE

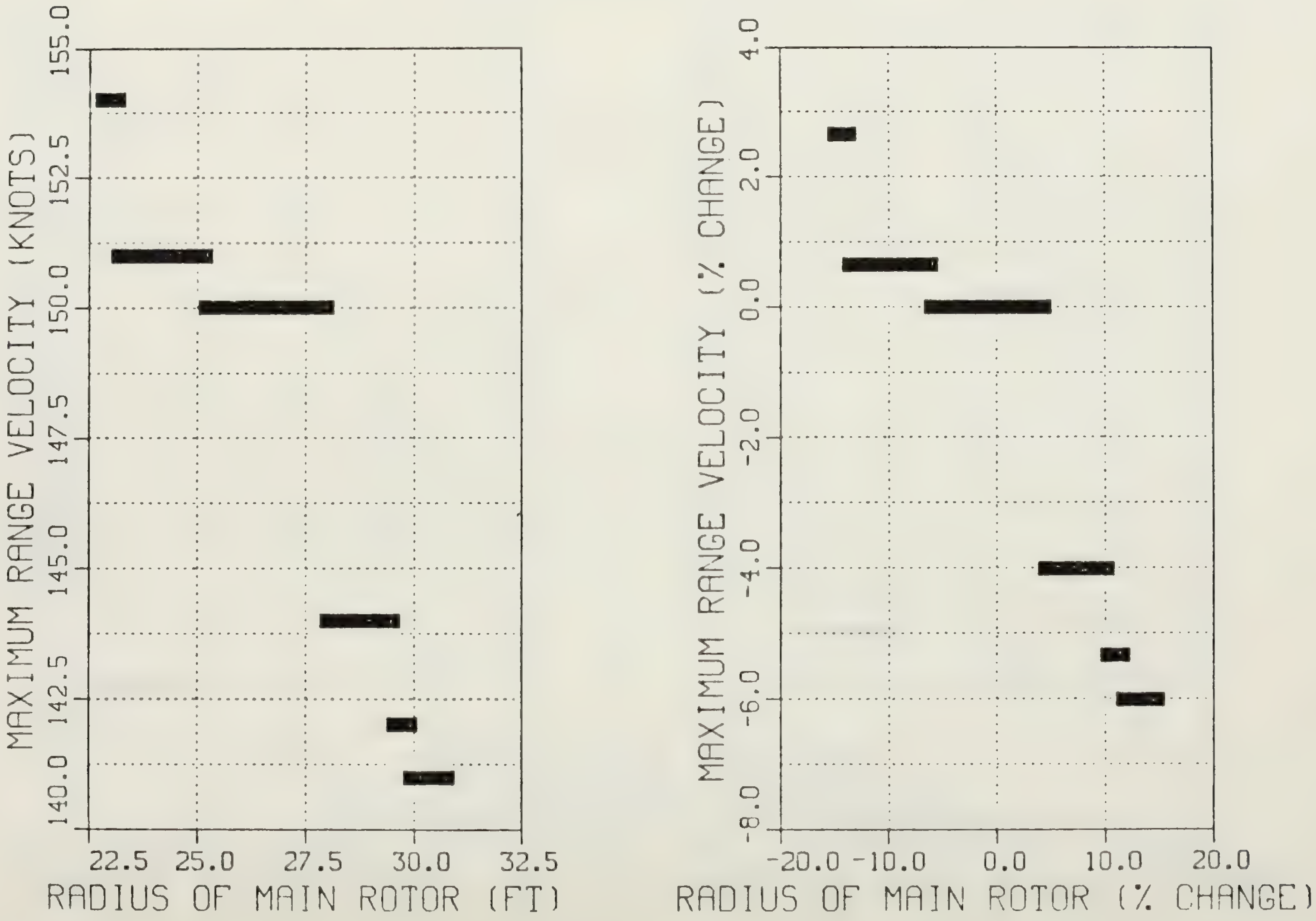
CHORD & ROTATIONAL VELOCITY HELD CONSTANT
SOLIDITY, TIP VELOCITY & ADVANCE RATIO
ALLOWED TO VARY WITH RADIUS





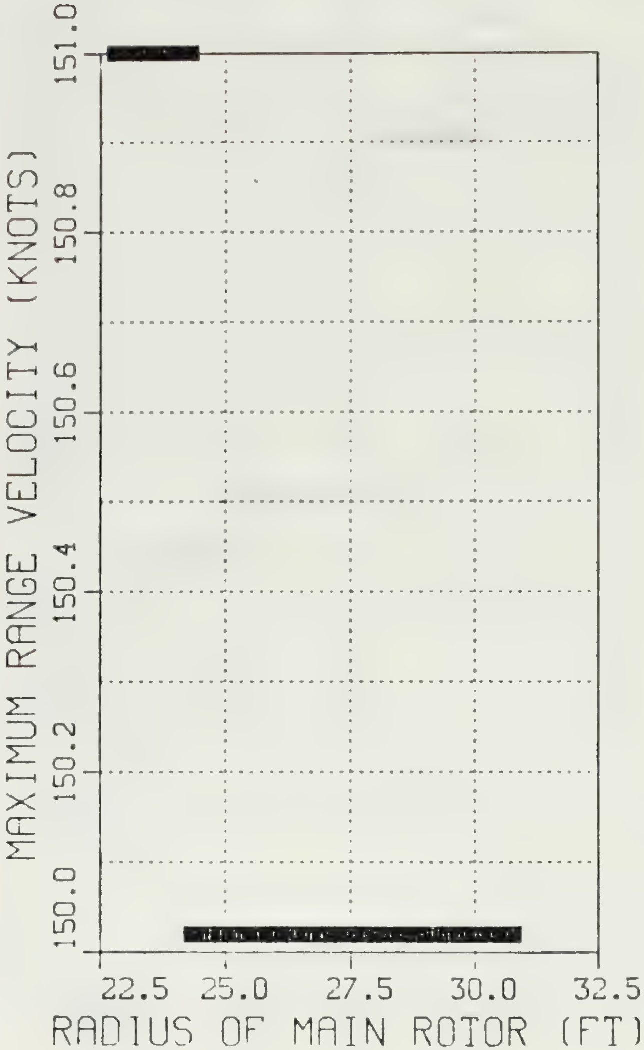
MAXIMUM RANGE VELOCITY VERSUS RADIUS CHANGE

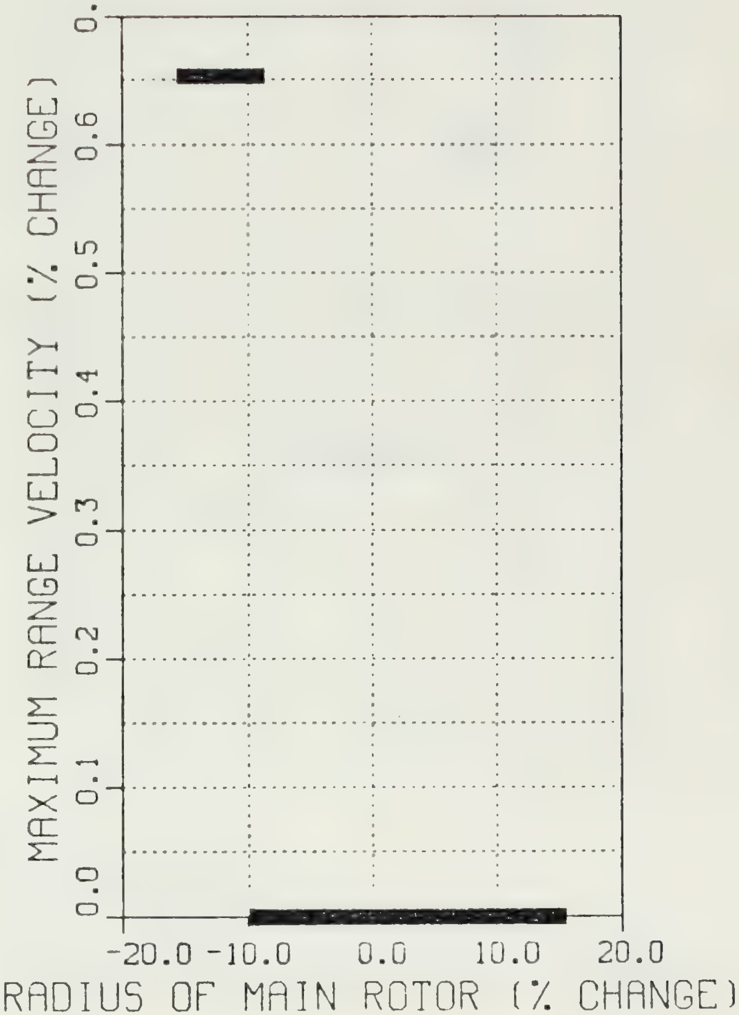
CHORD, TIP VELOCITY & ADVANCE RATIO HELD CONSTANT
SOLIDITY, ROTATIONAL VELOCITY
ALLOWED TO VARY WITH RADIUS



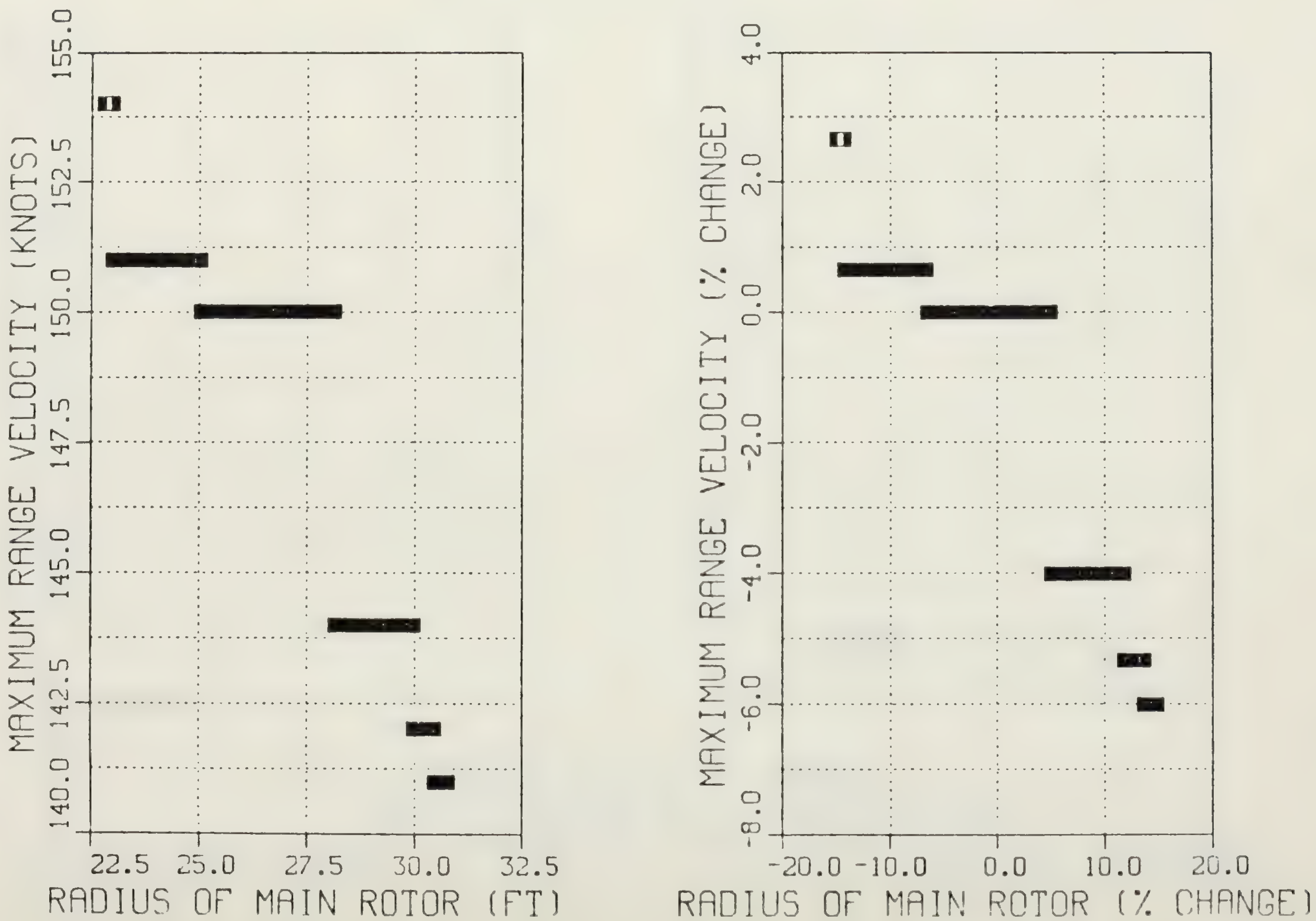
MAXIMUM RANGE VELOCITY VERSUS RADIUS CHANGE

SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT
CHORD,ADVANCE RATIO & TIP VELOCITY
ALLOWED TO VARY WITH RADIUS

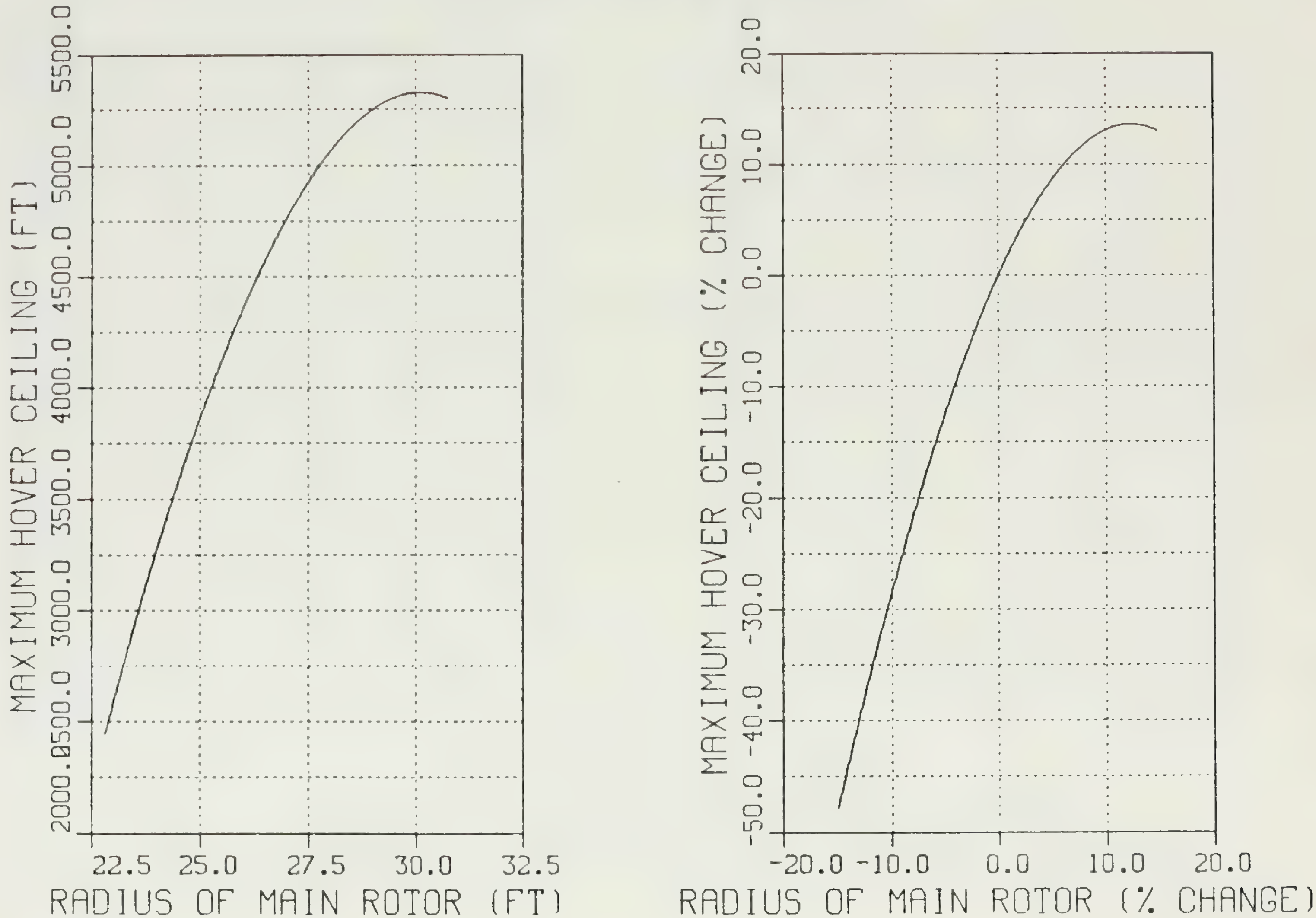




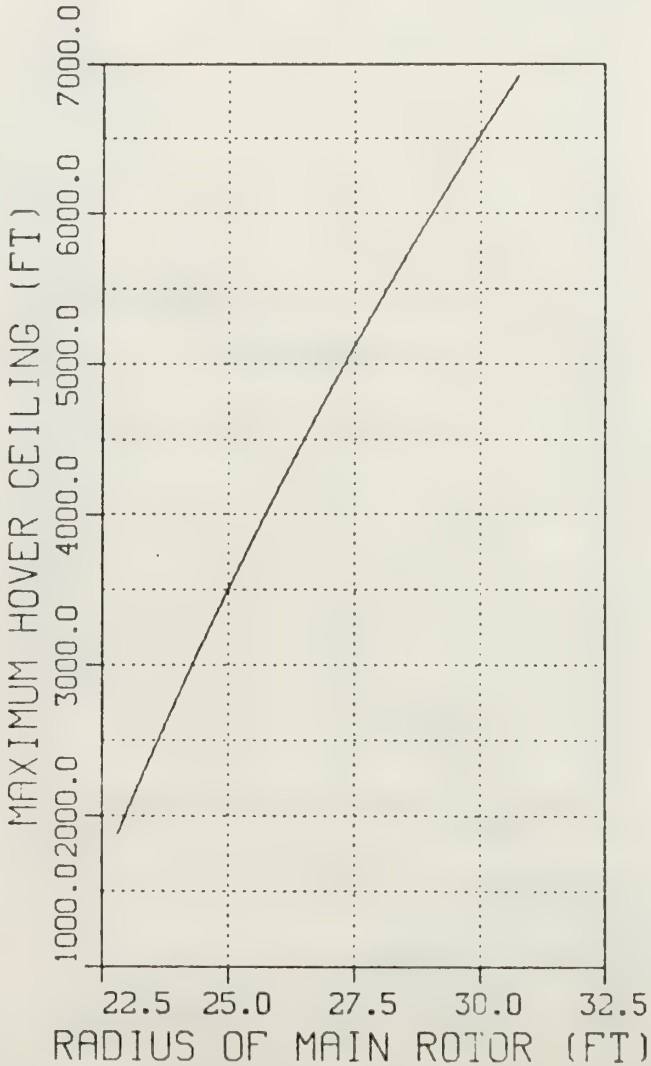
MAXIMUM RANGE VELOCITY VERSUS RADIUS CHANGE
SOLIDITY, ADVANCE RATIO & TIP VELOCITY HELD CONSTANT
CHORD & ROTATIONAL VELOCITY
ALLOWED TO VARY WITH RADIUS

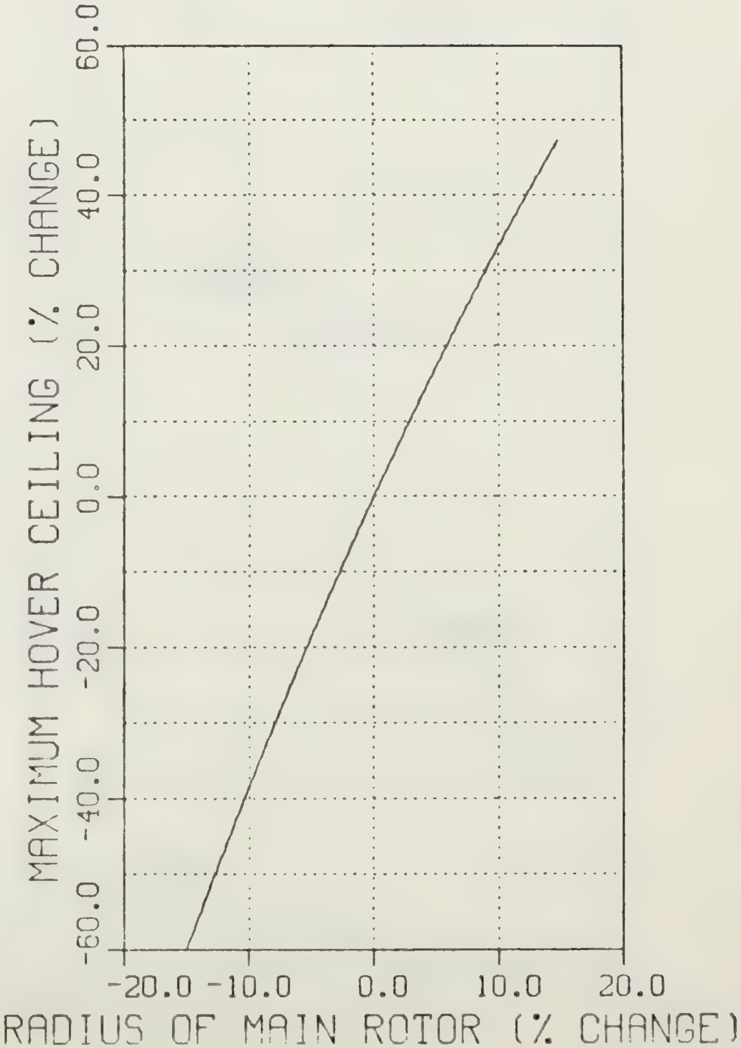


MAXIMUM HOVER CEILING (IGE) VERSUS RADIUS
CHORD & ROTATIONAL VELOCITY HELD CONSTANT
SOLIDITY, TIP VELOCITY & ADVANCE RATIO
ALLOWED TO VARY WITH RADIUS

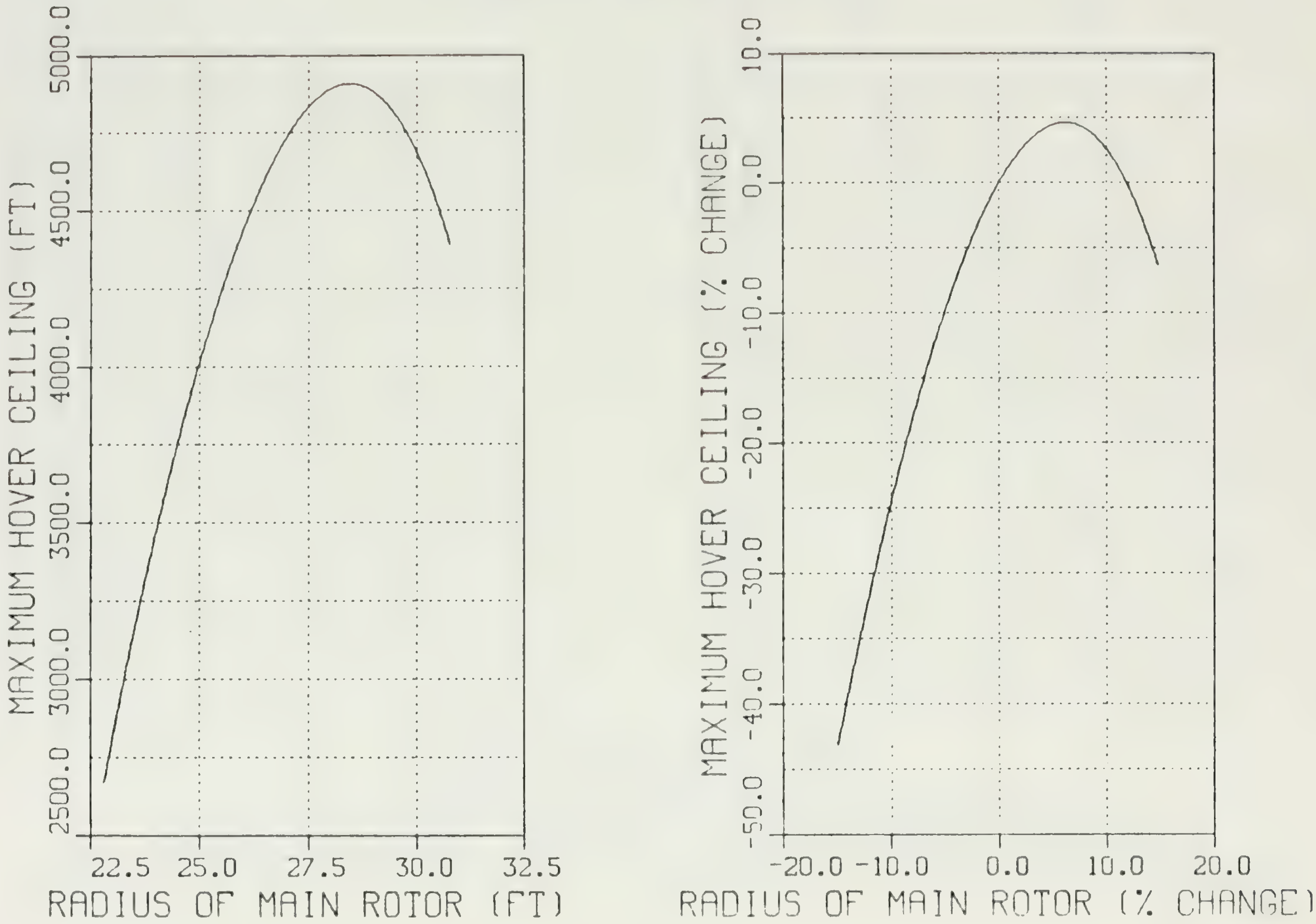


MAXIMUM HOVER CEILING (IGE) VERSUS RADIUS
CHORD,TIP VELOCITY & ADVANCE RATIO HELD CONSTANT
SOLIDITY, ROTATIONAL VELOCITY
ALLOWED TO VARY WITH RADIUS



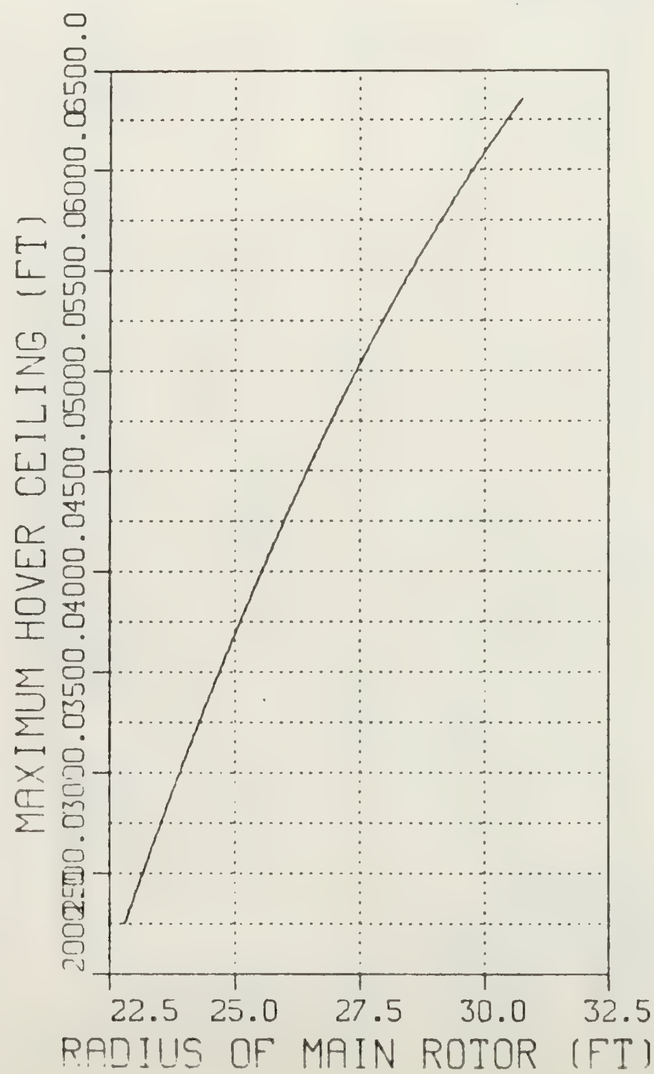


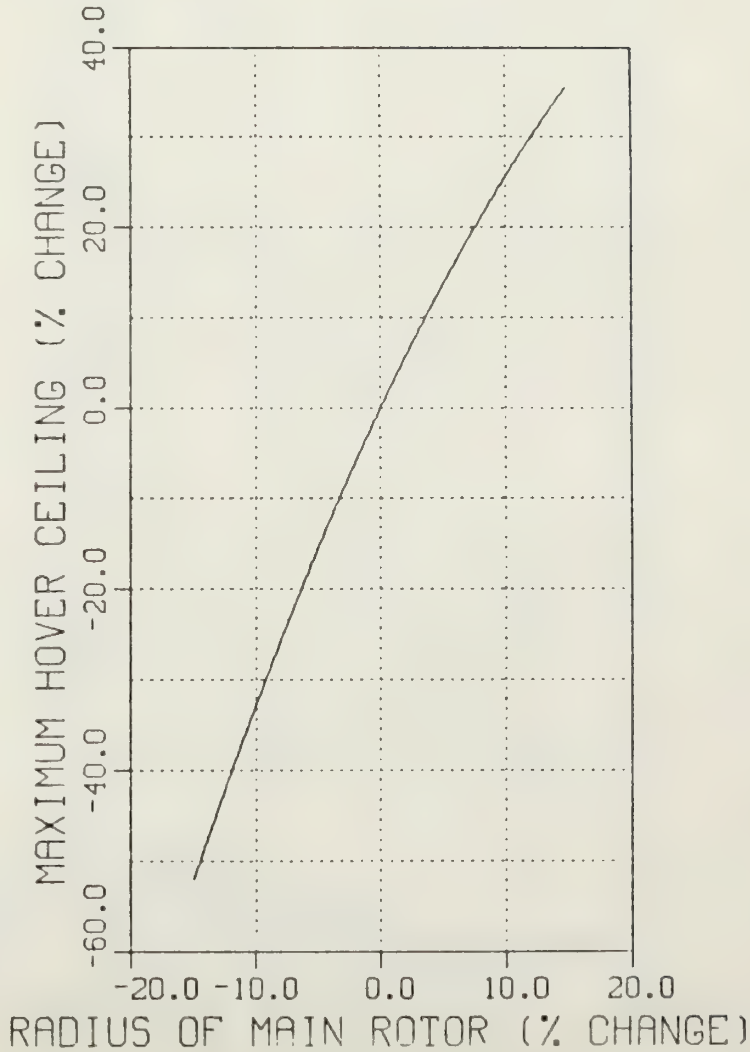
MAXIMUM HOVER CEILING (IGE) VERSUS RADIUS
SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT
CHORD, ADVANCE RATIO & TIP VELOCITY
ALLOWED TO VARY WITH RADIUS



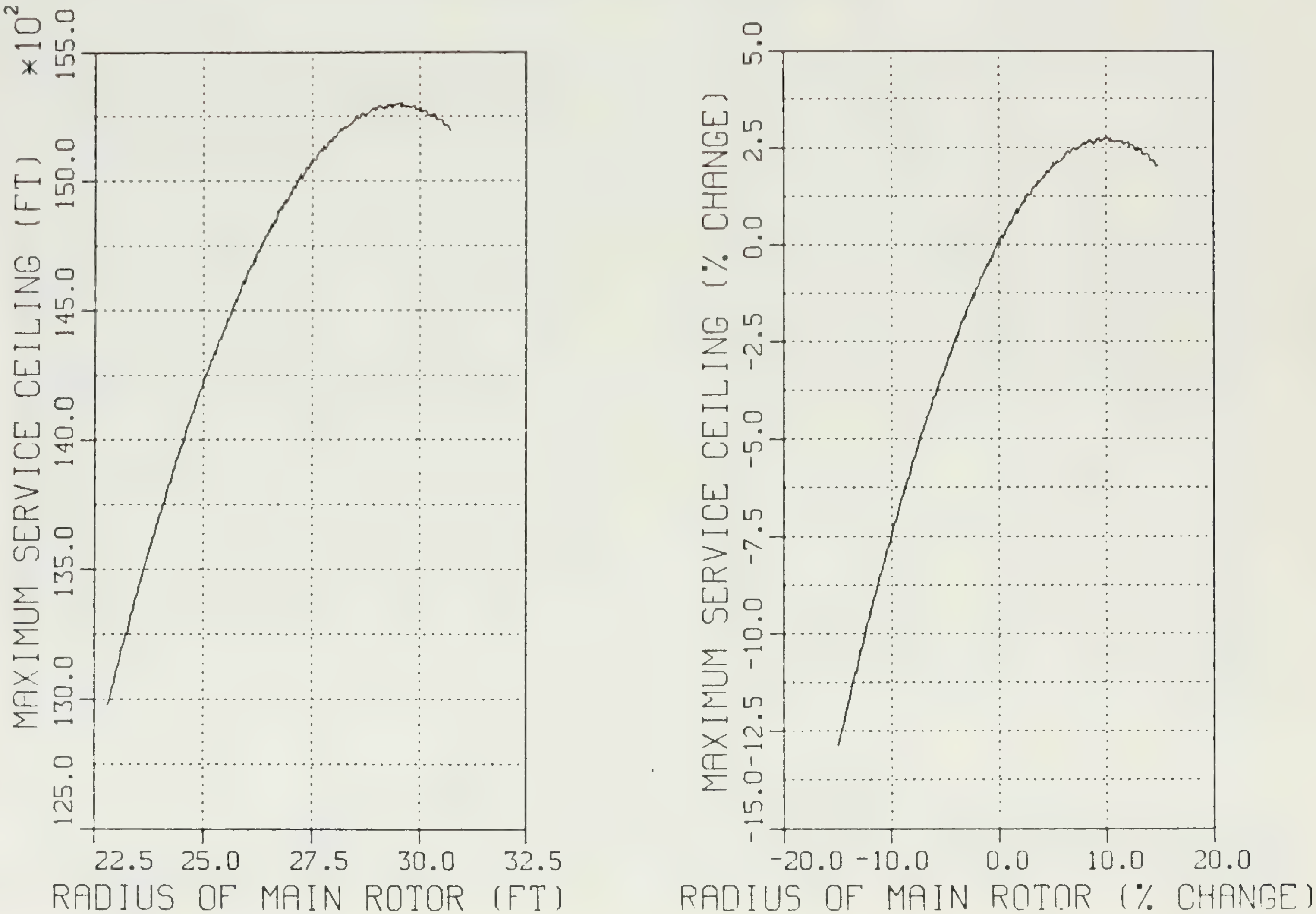
MAXIMUM HOVER CEILING (IGE) VERSUS RADIUS

SOLIDITY,ADVANCE RATIO & TIP VELOCITY HELD CONSTANT
CHORD & ROTATIONAL VELOCITY
ALLOWED TO VARY WITH RADIUS



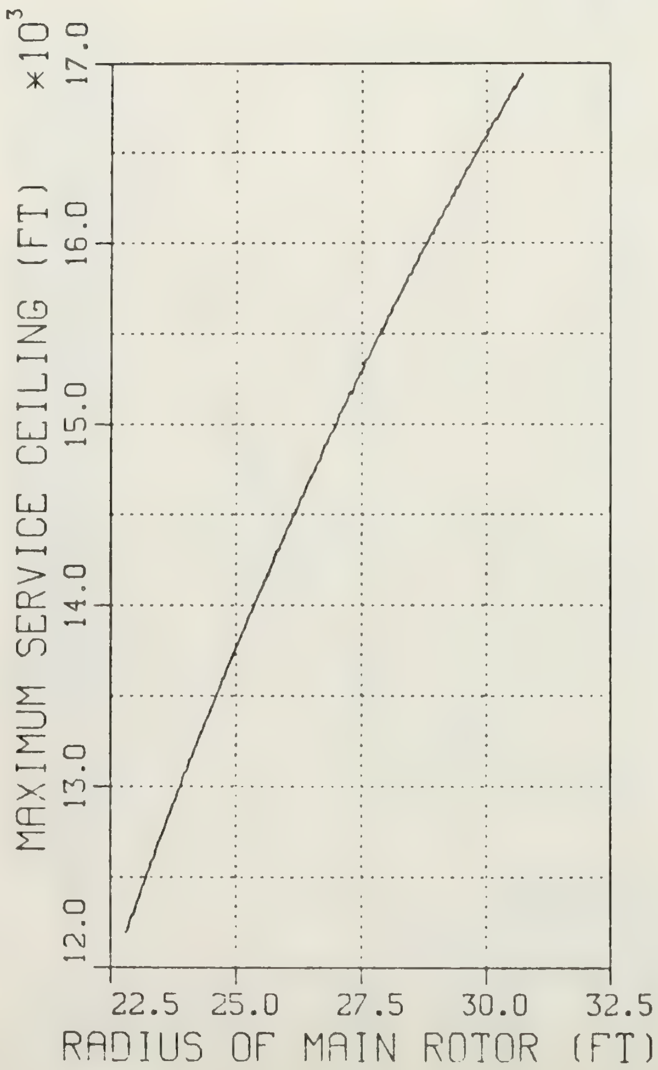


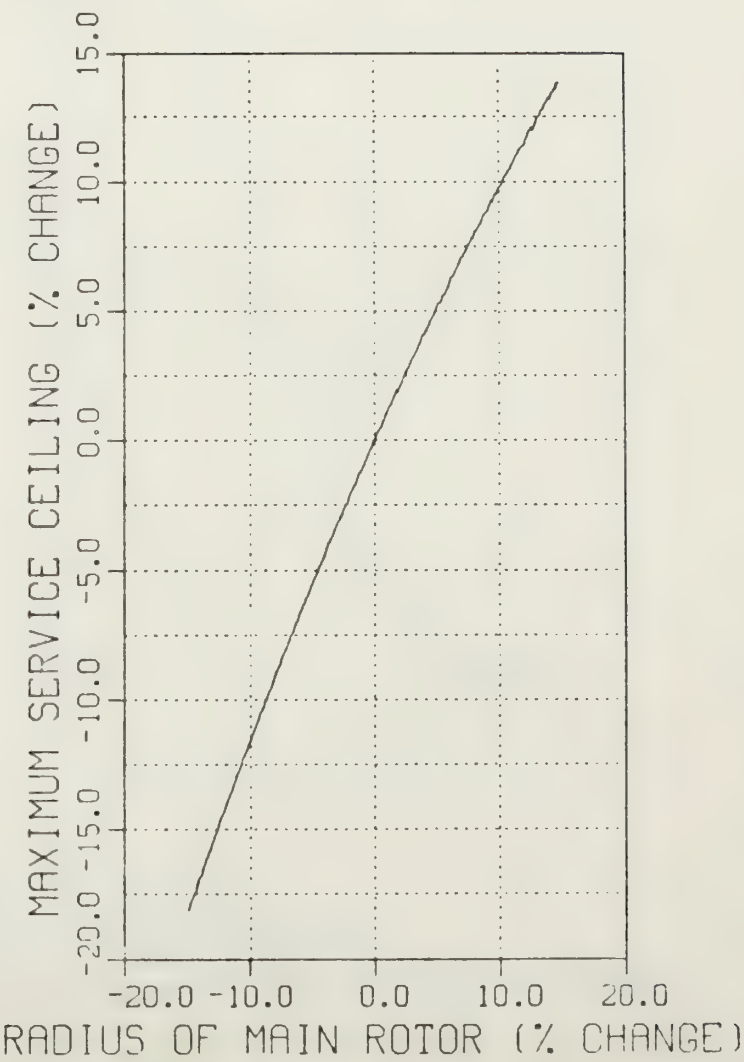
MAXIMUM SERVICE CEILING VERSUS RADIUS CHANGE
CHORD & ROTATIONAL VELOCITY HELD CONSTANT
SOLIDITY, TIP VELOCITY & ADVANCE RATIO
ALLOWED TO VARY WITH RADIUS



MAXIMUM SERVICE CEILING VERSUS RADIUS CHANGE

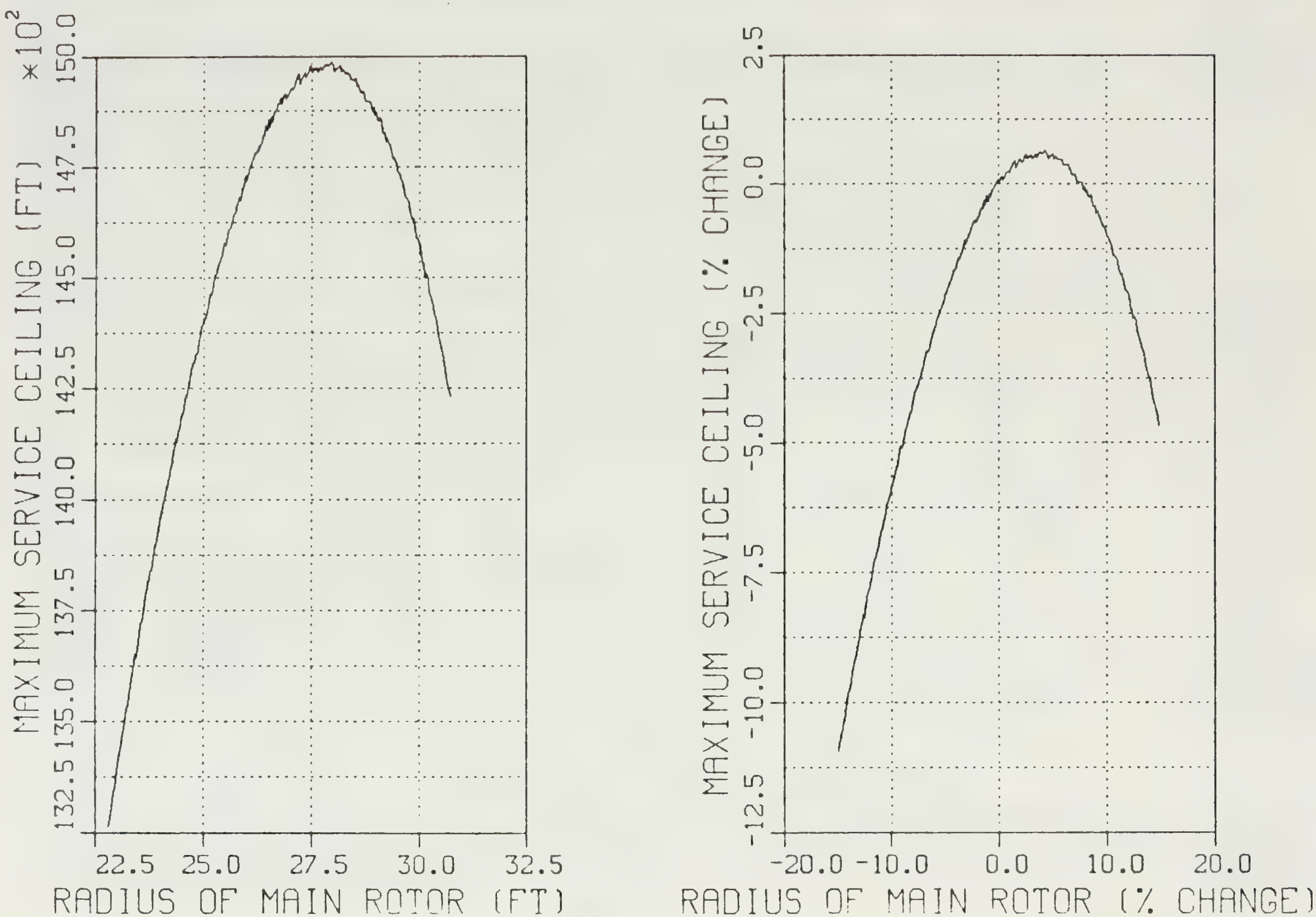
CHORD,TIP VELOCITY & ADVANCE RATIO HELD CONSTANT
SOLIDITY, ROTATIONAL VELOCITY
ALLOWED TO VARY WITH RADIUS





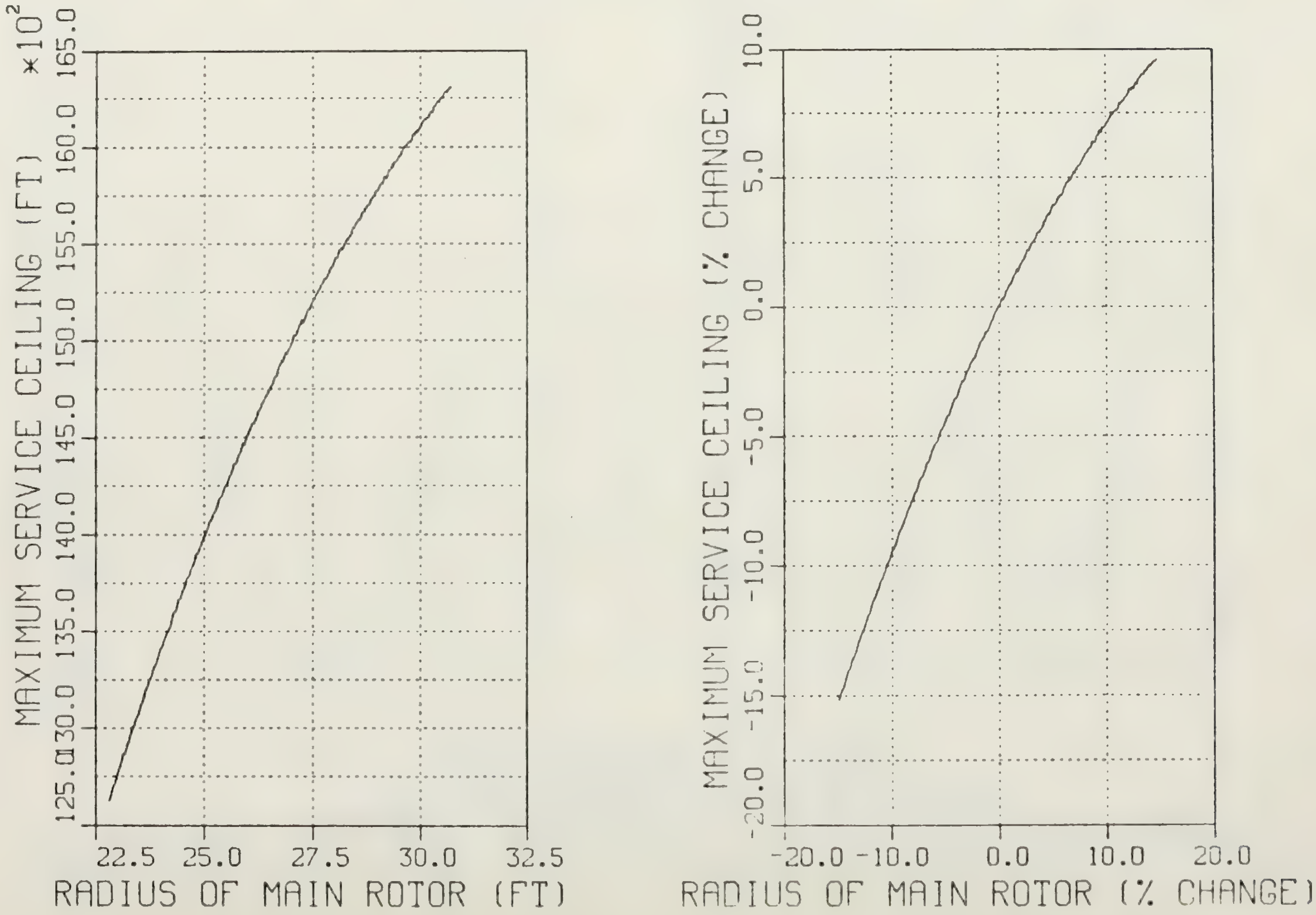
MAXIMUM SERVICE CEILING VERSUS RADIUS CHANGE

SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT
CHORD, ADVANCE RATIO & TIP VELOCITY
ALLOWED TO VARY WITH RADIUS



MAXIMUM SERVICE CEILING VERSUS RADIUS CHANGE

SOLIDITY,ADVANCE RATIO & TIP VELOCITY HELD CONSTANT
CHORD & ROTATIONAL VELOCITY
ALLOWED TO VARY WITH RADIUS



APPENDIX B

COMPUTER PROGRAM LISTING

```
C                     PERFORMANCE DESIGN PROGRAM
C
C                      COMPUTER PROGRAM LISTING
C
C
C*******************************************************************
C
C         THIS PROGRAM WILL TAKE A SET OF HELICOPTER DESIGN DATA,
C     CALCULATE POWER REQUIRED, ENDURANCE VELOCITY,RATE OF CLIMB, RANGE
C     VELOCITY, HOVER CEILING AND SERVICE CEILING FOR THAT DATA.  IT
C     ALSO WILL CALCULATE THE PERFORMANCE FACTORS LISTED ABOVE OVER A
C     RANGE OF VALUES FOR THE GIVEN DATA.  IT THEN WILL PLOT THE RESULTS
C     VERSUS MAIN ROTOR RADIUS IN TWO FORMS. FORM 1 BEING THE PERFORMANCE
C     FACTOR VALUE VERSUS THE MAIN ROTOR RADIUS.  FORM 2 BEING A
C     PERCENT DIFFERENCE OF THE PERFORMANCE VALUE FROM THE VALUE OF THE
C     FACTOR AT THE GIVEN SPECIFICATION VALUE VERSUS THE PERCENT
C     DIFFERENCE OF THE RADIUS TO THE SPECIFICATION RADIUS.
C
C-------------------------------------------------------------------
C     TO RUN THE PROGRAM FOLLOW THESE STEPS:
C
C     1.  INPUT THE REQUIRED DATA IN LINES  173-207
C     2.  COMPILE PROGRAM USING: FORTHX PERFORM ( NOMAP
C         NOTE: IF YOU DO NOT USE ( NOMAP, YOU WILL NOT HAVE ENOUGH DISK
C         SPACE TO RUN THE PROGRAM
C     3.  GRAPH PROGRAM USING: DISSPLA PERFORM
C     4.  EXAMINE RESULTS BY USING: DISSPCP
C
C-------------------------------------------------------------------
C   ****************   DEFINITION OF TERMS   *********************
C
C      AHAT - ZERO HORSEPOWER FUEL FLOW INTERCEPT
C      AMR - DISK AREA OF THE MAIN ROTOR (FT**2)
C      ARMR - ADVANCE RATIO OF THE MAIN ROTOR SYSTEM
C      ARTR - ADVANCE RATIO OF THE TAIL ROTOR SYSTEM
C      ATR - DISK AREA OF THE TAIL ROTOR (FT**2)
C      BHAT - AVERAGE SLOPE OF THE FUEL FLOW VS HORSEPOWER
C      CDCMR - AVERAGE COEFFICIENT OF DRAG OF THE MAIN ROTOR BLADES
C      CDCTR - AVERAGE COEFFICIENT OF DRAG OF THE TAIL ROTOR BLADES
C      CEIL - SUBROUTINE TO CALCULATE MAX SERVICE CEILING
C      CMR,CMR1 - CHORD LENGHT OF MAIN ROTOR (FT)
C      CONST - SUBROUTINE TO CALCULATE COMMON USED CONSTANTS
C      CTMR - COEFFICIENT OF THRUST OF THE MAIN ROTOR SYSTEM
C      CTR - CHORD LENGHT OF TAIL ROTOR (FT)
C      CTTR - COEFFICIENT OF THRUST OF THE TAIL ROTOR SYSYTEM
C      D.... - USED TO INDICATE A DIFFERENCE IN A VARIABLE, SEE VARIABLE
C      DA - DENSITY ALTITUDE (FT)
C      DELTA - PRESSURE RATIO (ALTITUDE/SSL)
```

```
C        DEN,DEN1 - DENSITY OF ATMOSPHERE (SLUGS/FT**3)                     PERO0490
C        EFPAF - EFFECTIVE FLAT PLATE AREA - FORWARD (FT**2)                PERO0500
C        EFPAV - EFFECTIVE FLATE PLATE AREA - VERTICAL (FT**2)              PERO0510
C        ENDUR - SUBROUTINE TO CALCULATE ENDURANCE VELOCITY                 PERO0520
C        ENGPOW - POWER AVAILABLE PER ENGINE (SHP)                          PERO0530
C        FMERT - FIGURE OF MERIT                                            PERO0540
C        GE - GROUND EFFECT FACTOR                                          PERO0550
C        HD - ROTOR HEIGHT TO ROTOR DIAMETER RATIO                          PERO0560
C        HOVALT - ALTITUDE FOR MAX HOVER CEILING (FT)                       PERO0570
C        HOVER - SUBROUTINE TO CALCULATE MAX HOVER CEILING                  PERO0580
C        IANS - PERFORMANCE FACTOR BEING CALCULATED                         PERO0590
C        IBMR - NUMBER OF BLADES OF THE MAIN ROTOR SYSTEM                   PERO0600
C        IBTR - NUMBER OF BLADES OF THE TAIL ROTOR SYSTEM                   PERO0610
C        IENG - NUMBER OF ENGINES                                           PERO0620
C        L - LENGHT OF TAIL BOOM (FT)                                       PERO0630
C        MAX... - MAXIMUM VALUE FOR X OR Y AXIS                             PERO0640
C        MAXRC - SUBROUTINE TO CALCULATE MAX RATE OF CLIMB                  PERO0650
C        MIN... - MINIMUM VALUE FOR X OR Y AXIS                             PERO0660
C        PA, PRESS1 - PRESSURE ALTITIDE (FT)                                PERO0670
C        PIMR - INDUCED POWER OF THE MAIN ROTOR WITH TIP LOSS & GROUND      PERO0680
C               EFFECT (SHP)                                                PERO0690
C        PITR - INDUCED POWER OF THE TAIL ROTOR WITH TIP LOSS (SHP)         PERO0700
C        PCMR - CLIMB POWER OF THE MAIN ROTOR (SHP)                         PERO0710
C        PSHP - PHANTOM SHAFT HORSEPOWER (SHP)                              PERO0720
C        POMR - PROFILE POWER OF THE MAIN ROTOR (SHP)                       PERO0730
C        POTR - PROFILE POWER OF THE TAIL ROTOR (SHP)                       PERO0740
C        POWER - SUBROUTINE TO CALCULATE POWER REQUIREMENTS                 PERO0750
C        POWMR - SUBROUTINE TO CALCULATE POWER REQUIREMENTS FOR MAIN ROTOR PERO0760
C        POWTR - SUBROUTINE TO CALCULATE POWER REQUIREMENTS FOR TAIL ROTOR PERO0770
C        PPMR - PARASITE POWER OF THE MAIN ROTOR (SHP)                      PERO0780
C        POWAA - POWER AVAILABLE AT ALTITUDE (SHP)                          PERO0790
C        PTAC - TOTAL POWER OF THE AIRCRAFT (SHP)                           PERO0800
C        PTAVAL - POWER AVAILABLE FOR AIRCRAFT                              PERO0810
C        PTMIN - MINIMUM POWER REQUIRED FOR MAXIMUM ENDURANCE               PERO0820
C        PTMR - TOTAL POWER OF THE MAIN ROTOR (SHP)                         PERO0830
C        PTTR - TOTAL POWER OF THE TAIL ROTOR (SHP)                         PERO0840
C        RANGE - SUBROUTINE TO CALCULATE THE MAX RANGE VELOCITY             PERO0850
C        RCMAX,RCMAXX - MAXIMUM RATE OF CLIMB (FT/MIN)                      PERO0860
C        RHC - SUBROUTINE TO CALCULATE DENSITY                              PERO0870
C        RMR,RRMR - RADIUS OF THE MAIN ROTOR (FT)                           PERO0880
C        RTR - RADIUS OF THE TAIL ROTOR (FT)                                PERO0890
C        RTHT - MAIN ROTOR HEIGHT ABOVE THE SKIDS OR WHEELS (FT)            PERO0900
C        RVMR,RVMR1 - ROTATIONAL VELOCITY OF THE MAIN ROTOR (RAD/SEC)       PERO0910
C        RVTR - ROTATIONAL VELOCITY OF THE TAIL ROTOR (RAD/SEC)             PERO0920
C        SDMR,SDMR3 - SOLIDITY OF THE MAIN ROTOR                            PERO0930
C        SDTR - SOLITITY OF THE TAIL ROTOR                                  PERO0940
C        SEV,SEVALT - ALTITUDE FOR SERVICE CEILING (FT)                     PERO0950
C        SFCM,SFCN,SFCC- SPECIFIC FUEL CONSUMPTION MILITARY,NORMAL,CRUISE  PERO0960
```

```
C       SHPM,SHPN,SHPC - SHAFT HORSEPOWER (SSL) MILITARY,NORMAL,CRUISE
C       SHPRAN - SHAFT HORSEPOWER REQUIRED AT RANGE VELOCITY (SHP)
C       SKDHT - SKID OR WHEEL HEIGHT ABOVE THE THE GROUND (FT)
C       TEMP,TEMP1 - TEMPERATURE (DEGREES F)
C       THETA - TEMPERATURE RATIO (ALTITUDE/SSL)
C       TLMR - TIP LOSS FACTOR OF THE MAIN ROTOR
C       TLTR - TIP LOSS FACTOR OF THE TAIL ROTOR
C       TTR - THRUST OF THE TAIL ROTOR (LBS FORCE)
C       VELMR - SUBROUTINE TO CALCULATE INDUCED VELOCITY FOR MAIN ROTOR
C       VELTR - SUBROUTINE TO CALCULATE INDUCED VELOCITY FOR TAIL ROTOR
C       VF,VFF,VF1 - FORWARD VELOCITY (KNOTS)
C       VFEND - VELOCITY FOR MAXIMUM ENDURANCE
C       VFRAN - RANGE VELOCITY (KTS)
C       VV - VERTICAL VELOCITY (FT/MIN)
C       VIHMR - INDUCED VELOCITY OF MAIN ROTOR AT A HOVER (FT/SEC)
C       VIHTR - INDUCED VELOCITY OF TAIL ROTOR AT A HOVER (FT/SEC)
C       VIMR - INDUCED VELOCITY OF MAIN ROTOR (FT/SEC)
C       VITR - INDUCED VELOCITY OF TAIL ROTOR (FT/SEC)
C       VITMR - INDUCED VELOCITY OF MAIN ROTOR IN FORWARD FLIGHT (FT/SEC)
C       VITTR - INDUCED VELOCITY OF TAIL ROTOR IN FORWARD FLIGHT (FT/SEC)
C       VIVMR - INDUCED VELOCITY OF MAIN ROTOR IN VERTICAL FLIGHT (FT/SEC)
C       VTMR, VTMR2 - TIP VELOCITY OF MAIN ROTOR (FT/SEC)
C       W - TOTAL WEIGHT OF THE HELICOPTER (LBS)
C       WDE,RANMAX - RATIO OF WEIGHT AND DRAG
C       WFM,WFN,WFC - FUEL FLOW RATE
C
C  *****************************************************************
C  *****************************************************************
C******************************************************************
        IMPLICIT REAL (A-H,L-Z)
        COMMON /W/ RCMAX(400),DRCMAX(400),VFRAN(400),DVFRAN(400),
     * HOVALT(400),DHOVAL(400),SEVALT(400),DSEVAL(400),FMERT(400),
     * DFMERT(400),VFEND(400),DVFEND(400)
        COMMON /A/ AMR,ARMR,ARTR,ATR,CDOMR,CDOTR,CMR,CTMR,CTR,CTTR,
     * DA,DEN,EFPAF,EFPAV,GE,HD,L,PA,RADIUS,RTR,RTHT,RVMR,RVTR,SDMR,
     * SDTR,SKDHT,TEMP,W,VF,VV,IBTR,IBMR,DENSSL,PIE,PTMIN(400),VIMR,
     * VITR,VTTR,VTMR,RANMAX,THETA,DELTA,SHPRAN(400),VV1,VF1,IENG,SDMR3,
     * VTMR2,DIFF,SFCM,SFCN,SFCC,SHPM,SHPN,SHPC,CMR1,RVMR1,RRMR
        COMMON /B/ PIMR(400),POMR(400),PPMR(400),PCMR(400),PITR(400),
     * PCTR(400),PTMR(400),PTTR(400),PTAC(400)
        COMMON /C/ RMR(400),I,J,K,DRMR(400),MINX,MAXX,MINY,MAXY,MINZ,
     * MAXZ,DMIN,DMAX,MINB,MAXB,MINC,MAXC,MIN,MAX,MAXA,IANS,TEXT(200)
        COMMON /Y/DPIMR(400),DPOMR(400),DPPMR(400),DPCMR(400),DPTMR(400),
     * DPITR(400),DPOTR(400),DPTAC(400),DFTTR(400)
        COMMON /Z/ VFEND1(400),VFEND2(400),VFEND3(400),VFEND4(400)
C-----------------------------------------------------------------
 5      CALL FRTCMS('CLRSCRN ')
        WRITE(5,6000)
```

```
      WRITE(5,6001)
      WRITE(5,6002)
      WRITE(5,6003)
      WRITE(5,6004)
      WRITE(5,6005)
      WRITE(5,6006)
      WRITE(5,6007)
      WRITE(5,6008)
      WRITE(5,6009)
      CALL RDINT(IANS)
      CALL FRTCMS('CLRSCRN ')
      IF(IANS .EQ. 3 .OR. IANS .EQ. 6) WRITE (5,6010)
      WRITE(5,6011)
 6000 FORMAT(10X,'YOU MAY CHOOSE ANY OF THE FOLLOWING PERFORMANCE ')
 6001 FORMAT(10X,'FACTORS TO EXAMINE.  EACH ONE COULD TAKE 2-5 MINUTES')
 6002 FORMAT(10X,'TO RUN (ON AN AVERAGE). ',//)
 6003 FORMAT(15X,'1.  POWER REQUIREMENTS')
 6004 FORMAT(15X,'2.  MAXIMUM ENDURANCE VELOCITY')
 6005 FORMAT(15X,'3.  MAXIMUM RATE OF CLIMB')
 6006 FORMAT(15X,'4.  MAXIMUM RANGE VELOCITY')
 6007 FORMAT(15X,'5.  MAXIMUM HOVER CEILING (IGE)')
 6008 FORMAT(15X,'6.  MAXIMUM SERVICE CEILING')
 6009 FORMAT(10X,//,'ENTER THE NUMBER (1,2,3,4,5,OR 6) YOU WISH TO EXAMI
     *NE')
 6010 FORMAT(5X,'THIS MAY TAKE SOME TIME IF THERE ARE ALOT OF USERS')
 6011 FORMAT(5X,'CALCULATING RESULTS')
C*****************************************************************
C
C##########  INPUT DATA FOR HELICOPTER TO BE EXAMINED  #############
C
C*****************************************************************
C---------------------- INPUT MAIN ROTOR DATA ----------------------
      RADIUS = 26.8
C---------------------- +/- VALUE FROM SPEC RADIUS      ------------
      DIFF = 4.0
      CMR1 = 1.75
      RVMR1 = 27.0
      CDCMR = 0.008
      IBMR = 4
C---------------------- INPUT TAIL ROTOR DATA ----------------------
      RTR = 5.5
      CTR = 0.81
      RVTR = 124.6
      CDCTR = 0.008
      IBTR = 4
C---------------------- INPUT GENERAL AIRFRAME DATA ----------------
      L = 31.5
      W = 20000.0
```

```
      EFPAF = 25.7                                                      PER01930
      EFPAV = 31.8                                                      PER01940
      RTHT = 11.2                                                       PER01950
C--------------------- INPUT ENGINE DATA ------------------------------PER01960
      IENG = 2                                                          PER01970
      SFCM = 0.46                                                       PER01980
      SFCN = 0.47                                                       PER01990
      SFCC = 0.51                                                       PER02000
      SHPM = 1561.0                                                     PER02010
      SHPN = 1318.0                                                     PER02020
      SHPC = 989.0                                                      PER02030
C------------ INPUT FLIGHT VELOCITY FOR INITIAL POWER CALCULATIONS --- PER02040
      VF1 = 90.0                                                        PER02050
C=========   END OF REQUIRED INPUT INFORMATION  ========================PER02060
C======================================================================PER02070
      DO 3000 K = 1,4                                                   PER02080
      VV1 = 0.0                                                         PER02090
      PA = 0.0                                                          PER02100
      TEMP = 0.0                                                        PER02110
      DA = 0.0                                                          PER02120
      DENSSL = 0.00237694                                               PER02130
      SKDHT = 2500.0                                                    PER02140
      VF = VF1                                                          PER02150
      VV = VV1                                                          PER02160
      CMR = CMR1                                                        PER02170
      RVMR = RVMR1                                                      PER02180
      PIE = 3.141593                                                    PER02190
      SDMR3 = IBMR*CMR1 / (PIE*RADIUS)                                  PER02200
      VTMR2 = RVMR1*RADIUS                                              PER02210
      CALL RHO                                                          PER02220
      CALL RAD(DIFF,RADIUS,RRMR,RMR,DRMR,MINZ,MAXZ,DMIN,DMAX)           PER02230
      IF(IANS .NE. 1) GO TO 2800                                        PER02240
      CALL POWER                                                        PER02250
      GO TO 3000                                                        PER02260
2800  IF(IANS .NE. 2) GO TO 2810                                        PER02270
      CALL ENDUR                                                        PER02280
      GO TO 3000                                                        PER02290
2810  IF(IANS .NE. 3) GO TO 2820                                        PER02300
      CALL ENDUR                                                        PER02310
      CALL MAXFC                                                        PER02320
      GO TO 3000                                                        PER02330
2820  IF(IANS .NE. 4) GO TO 2830                                        PER02340
      CALL RANGE                                                        PER02350
      GO TO 3000                                                        PER02360
2830  IF(IANS .NE. 5) GO TO 2840                                        PER02370
      CALL HOVER                                                        PER02380
      GO TO 3000                                                        PER02390
2840  IF(IANS .NE. 6) GO TO 2850                                        PER02400
```

```
      CALL ENCUR
      CALL CEIL
      GO TO 3C00
2850  GO TO 5
3000  CONTINUE
      CALL DCNEPL
5000  CONTINUE
      STCP
      END
C*****************************************************************
C
C                   SUBROUTINES USED IN PRGRAM FCLLCW
C
C*****************************************************************
C*****************************************************************
      SUBROUTINE RAD(DIFF,RADIUS,RRMR,RMR,DRMR,MINZ,MAXZ,DMIN,DMAX)
      IMPLICIT REAL (A-H,L-Z)
      DIMENSICN RMR(400),CRMR(400)
      RRMR = RADILS - DIFF
      DO 10 I = 1,324
         RMR(I) = RRMR
         IF (I .EQ. 163) RMR(I) = RADIUS
         RRMR = RRMR + CIFF/162
 10   CONTINLE
      DO 20 I = 1,324
         DRMF(I) = ((RMR(I) - RMR(163)) / RMR(163)) * 100.0
 20   CONTINLE
         DMIN = DRMR(1)
         DMAX = DRMR(324)
         MINZ = RMR(1)
         MAXZ = RMR(324)
      RETURN
      END
C================================================================
C=============                           =========================
C===================       SUB-GROUP B       =======================
C=============    PERFCRMANCE SUBROUTINES    ======================
C==================                        ========================
C=================================================================
C*****************************************************************
C
C   SUBROUTINE TO CALCULATE ALL PARTS OF THE TOTAL POWER REQUIREMENTS
C        AND FIGURE OF MERIT.
C
C*****************************************************************
C*****************************************************************
      SUBROUTINE POWER
      IMPLICIT REAL (A-H,L-Z)
```

```
      INTEGER I,J,K
      COMMON /A/ AMR,ARMR,ARTR,ATR,CDOMR,CDOTR,CMR,CTMR,CTR,CTTR,
     * DA,DEN,EFPAF,EFPAV,GE,HD,L,PA,RADILS,RTR,RTHT,RVMR,RVTR,SDMR,
     * SDTR,SKCHT,TEMP,W,VF,VV,IBTR,IBMR,DENSSL,PIE,PTMIN(400),VIMR,
     * VITR,VTTR,VTMR,RANMAX,THETA,DELTA,SHPRAN(400),VV1,VF1,IENG,SDMR3,
     * VTMR2,DIFF,SFCM,SFCN,SFCC,SHPM,SHFN,SFFC,CMR1,RVMR1,RRMR
      COMMON /B/ PIMR(400),POMR(400),PPMR(400),PCMR(400),PITR(400),
     * POTR(400),PTMR(400),PTTR(400),PTAC(400)
      COMMON /C/ RMR(400),I,J,K,DRMR(400),MINX,MAXX,MINY,MAXY,MINZ,
     * MAXZ,CMIN,DMAX,MINB,MAXB,MINC,MAXC,MIN,MAX,MAXA,IANS,TEXT(200)
      COMMON /W/ RCMAX(400),DRCMAX(400),VFRAN(400),DVFRAN(400),
     * HOVALT(400),DHOVAL(400),SEVALT(400),DSEVAL(400),FMERT(400),
     * DFMERT(400),VFEND(400),DVFEND(400)
      COMMON /Y/DPIMR(400),DPOMR(400),DPPMR(400),DPCMR(400),DPTMR(400),
     * DPITR(400),DPOTR(400),DPTAC(400),DPTTR(400)
C-----------------------------------------------------------------------
      MAX = 0.0
      MAXA = 0.0
      DO 40 I = 1,324
         CALL CONST
         AMR = RMR(I)**2.0*PIE
         IF(K .NE.1) GO TO 8
             VTMR = RVMR*RMR(I)
             ARMR = VF / (RVMR*RMR(I))
             SDMR = IBMR*CMR / (PIE*RMR(I))
 8       IF(K .NE. 2) GO TO 10
             RVMR = VTMR2/RMR(I)
             VTMR = RVMR*RMR(I)
             ARMR = VF / (RVMR*RMR(I))
             SDMR = IBMR*CMR1 / (PIE*RMR(I))
 10      IF(K .NE. 3) GO TO 12
             CMR = SDMR3*PIE*RMR(I)/IBMR
             SDMR = IBMR*CMR / (PIE*RMR(I))
             RVMR = RVMR1
             ARMR = VF / (RVMR*RMR(I))
             VTMR = RVMR*RMR(I)
 12      IF(K .NE. 4) GO TO 14
             CMR = SDMR3*PIE*RMR(I)/IBMR
             SDMR = IBMR*CMR / (PIE*RMR(I))
             RVMR = VTMR2/RMR(I)
             VTMR = RVMR*RMR(I)
             ARMR = VF / (RVMR*RMR(I))
 14      CALL VELMR
         CALL FCWMR
         CALL VELTR
         CALL FCWTR
         IF(PIMR(I) .GT. MAXA) MAXA = PIMR(I)
         IF(POMR(I) .GT. MAXA) MAXA = POMR(I)
```

```
      IF(PPMR(I) .GT. MAXA) MAXA = PPMR(I)                              PER03370
      IF(VV .LE. 0.0 ) GO TO 20                                         PER03380
      IF(PCMR(I) .GT. MAXA) MAXA = PCMR(I)                              PER03390
20    IF(PTAC(I) .GT. MAX) MAX = PTAC(I)                                PER03400
      VF = VF / 1.68889                                                 PER03410
      VV = VV * 60.0                                                    PER03420
40    CONTINUE                                                          PER03430
C-----------------------------------------------------------------------PER03440
      MINY = 500.0                                                      PER03450
      MAXY = -500.0                                                     PER03460
      MINX = 500.0                                                      PER03470
      MAXX = -500.0                                                     PER03480
      DO 140 I = 1,324                                                  PER03490
         DPIMR(I) = ((PIMR(I) - PIMR(163)) / PIMR(163)) * 100.0         PER03500
           IF(DPIMR(I) .GT. MAXX) MAXX = DPIMR(I)                       PER03510
           IF(DPIMR(I) .LT. MINX) MINX = DPIMR(I)                       PER03520
         DPOMR(I) = ((POMR(I) - POMR(163)) / POMR(163)) * 100.0         PER03530
           IF(DPOMR(I) .GT. MAXX) MAXX = DPOMR(I)                       PER03540
           IF(DPOMR(I) .LT. MINX) MINX = DPOMR(I)                       PER03550
         DPPMR(I) = ((PPMR(I) - PPMR(163)) / PPMR(163)) * 100.0         PER03560
           IF(DPPMR(I) .GT. MAXX) MAXX = DPPMR(I)                       PER03570
           IF(DPPMR(I) .LT. MINX) MINX = DPPMR(I)                       PER03580
         IF(VV .LE. 0.0 ) GO TO 120                                     PER03590
         DPCMR(I) = ((PCMR(I) - PCMR(163)) / PCMR(163)) * 100.0         PER03600
           IF(DPCMR(I) .GT. MAXX) MAXX = DPCMR(I)                       PER03610
           IF(DPCMR(I) .LT. MINX) MINX = DPCMR(I)                       PER03620
120      DPTMR(I) = ((PTMR(I) - PTMR(163)) / PTMR(163)) * 100.0         PER03630
         DPITR(I) = ((PITR(I) - PITR(163)) / PITR(163)) * 100.0         PER03640
         DPOTR(I) = ((POTR(I) - POTR(163)) / POTR(163)) * 100.0         PER03650
         DPTTR(I) = ((PTTR(I) - PTTR(163)) / PTTR(163)) * 100.0         PER03660
         DPTAC(I) = ((PTAC(I) - PTAC(163)) / PTAC(163)) * 100.0         PER03670
          IF(DPTAC(I) .GT. MAXY) MAXY = DPTAC(I)                        PER03680
          IF(DPTMR(I) .GT. MAXY) MAXY = DPTMR(I)                        PER03690
          IF(DPTTR(I) .GT. MAXY) MAXY = DPTTR(I)                        PER03700
          IF(DPTAC(I) .LT. MINY) MINY = DPTAC(I)                        PER03710
          IF(DPTMR(I) .LT. MINY) MINY = DPTMR(I)                        PER03720
          IF(DPTTR(I) .LT. MINY) MINY = DPTTR(I)                        PER03730
140   CONTINUE                                                          PER03740
      CALL GRAF1C                                                       PER03750
      CALL GRAF1D                                                       PER03760
C-----------------------------------------------------------------------PER03770
      VF = 0.0                                                          PER03780
      MINB = 1.0                                                        PER03790
      MAXB = 0.0                                                        PER03800
      DO 150 I = 1,324                                                  PER03810
         CALL CONST                                                     PER03820
         AMR = RMR(I)**2.0*PIE                                          PER03830
         IF(K .NE.1) GO TO 142                                          PER03840
```

```
            VTMR =  RVMR*RMR(I)                                          PER03850
            ARMR =  VF / (RVMR*RMR(I))                                   PER03860
            SDMR =  IBMR*CMR / (PIE*RMR(I))                              PER03870
142      IF(K .NE. 2) GO TO 144                                          PER03880
            RVMR =  VTMR2/RMR(I)                                         PER03890
            VTMR =  RVMR*RMR(I)                                          PER03900
            ARMR =  VF / (RVMR*RMR(I))                                   PER03910
            SDMR =  IBMR*CMR1 / (PIE*RMR(I))                             PER03920
144      IF(K .NE. 3) GO TO 146                                          PER03930
            CMR = SDMR3*PIE*RMR(I)/IBMR                                  PER03940
            SDMR =  IBMR*CMR / (PIE*RMR(I))                              PER03950
            RVMR =  RVMR1                                                PER03960
            ARMR =  VF / (RVMR*RMR(I))                                   PER03970
            VTMR =  RVMR*RMR(I)                                          PER03980
146      IF(K .NE. 4) GO TO 148                                          PER03990
            CMR = SDMR3*PIE*RMR(I)/IBMR                                  PER04000
            SDMR =  IBMR*CMR / (PIE*RMR(I))                              PER04010
            RVMR =  VTMR2/RMR(I)                                         PER04020
            VTMR =  RVMR*RMR(I)                                          PER04030
            ARMR =  VF / (RVMR*RMR(I))                                   PER04040
148      CALL VELMR                                                      PER04050
         CALL POWMR                                                      PER04060
         CALL VELTR                                                      PER04070
         CALL FCWTR                                                      PER04080
         FMERT(I) = PIMR(I) / (PIMR(I) + PCMR(I))                        PER04090
         IF(FMERT(I) .GT. MAXB) MAXB = FMERT(I)                          PER04100
         IF(FMERT(I) .LT. MINB) MINB = FMERT(I)                          PER04110
150     CONTINUE                                                         PER04120
C------------------------------------------------------------------------PER04130
       MINC = 500.0                                                      PER04140
       MAXC = -500.0                                                     PER04150
       DO 155 I = 1,324                                                  PER04160
          DFMERT(I) = ((FMERT(I) - FMERT(163)) / FMERT(163)) * 100.0     PER04170
            IF(DFMERT(I) .GT. MAXC) MAXC = DFMERT(I)                     PER04180
            IF(DFMERT(I) .LT. MINC) MINC = DFMERT(I)                     PER04190
155    CONTINUE                                                          PER04200
          CALL GRAFIE                                                    PER04210
       RETURN                                                            PER04220
       END                                                               PER04230
C*********************************************************************** PER04240
C                                                                        PER04250
C      SUBROUTINE TO CALCULATE THE ENDURANCE  VELOCITY                   PER04260
C                                                                        PER04270
C*********************************************************************** PER04280
C*********************************************************************** PER04290
       SUBROUTINE ENDUR                                                  PER04300
       IMPLICIT REAL (A-H,L-Z)                                           PER04310
       COMMON /A/ AMR,ARMR,ARTR,ATR,CDCMR,CDOTR,CMR,CTMR,CTR,CTTR,       PER04320
```

```
     * DA,DEN,EFPAF,EFPAV,GE,HD,L,PA,RADILS,RTR,RTHT,RVMR,RVTR,SCMR,      PER04330
     * SCTR,SKCHT,TEMP,W,VF,VV,IBTR,IBMR,CENSSL,PIE,PTMIN(400),VIMR,      PER04340
     * VITR,VTTR,VTMR,RANMAX,THETA,DELTA,SHPRAN(400),VV1,VF1,IENG,SDMR3,  PER04350
     * VTMR2,DIFF,SFCM,SFCN,SFCC,SHPM,SHPN,SHPC,CMR1,RVMR1,RRMR           PER04360
       COMMON /B/ PIMR(400),POMR(400),PPMR(400),PCMR(400),PITR(400),      PER04370
     * POTR(400),PTMR(400),PTTR(400),PTAC(400)                            PER04380
       COMMON /C/ RMR(400),I,J,K,DRMR(400),MINX,MAXX,MINY,MAXY,MINZ,      PER04390
     * MAXZ,DMIN,DMAX,MINB,MAXB,MINC,MAXC,MIN,MAX,MAXA,IANS,TEXT(200)     PER04400
      COMMON /W/ RCMAX(400),DRCMAX(400),VFRAN(400),DVFRAN(400),           PER04410
     * HCVALT(400),DHOVAL(400),SEVALT(400),DSEVAL(400),FMERT(400),        PER04420
     * DFMERT(400),VFEND(400),DVFEND(400)                                 PER04430
       COMMON /Z/ VFEND1(400),VFEND2(400),VFEND3(400),VFEND4(400)         PER04440
C-------------------------------------------------------------------------PER04450
       MAX = 0.0                                                          PER04460
       MIN = 500.0                                                        PER04470
       CALL RHC                                                           PER04480
       DO 280 I = 1,324                                                   PER04490
          AMR = RMR(I)**2.0*PIE                                           PER04500
          IF(K .NE.1) GO TO 148                                           PER04510
             RVMR = RVMR1                                                 PER04520
             VTMR = RVMR*RMR(I)                                           PER04530
             SCMR = IBMR*CMR / (PIE*RMR(I))                               PER04540
148       IF(K .NE. 2) GO TO 150                                          PER04550
             RVMR = VTMR2/RMR(I)                                          PER04560
             VTMR = RVMR*RMR(I)                                           PER04570
             SCMR = IBMR*CMR1 / (PIE*RMR(I))                              PER04580
150       IF(K .NE. 3) GO TO 152                                          PER04590
             CMR = SDMR3*PIE*RMR(I)/IBMR                                  PER04600
             SCMR = IBMR*CMR / (PIE*RMR(I))                               PER04610
             RVMR = RVMR1                                                 PER04620
             VTMR = RVMR*RMR(I)                                           PER04630
152       IF(K .NE. 4) GO TO 154                                          PER04640
             CMR = SDMR3*PIE*RMR(I)/IBMR                                  PER04650
             SCMR = IBMR*CMR / (PIE*RMR(I))                               PER04660
             RVMR = VTMR2/RMR(I)                                          PER04670
             VTMR = RVMR*RMR(I)                                           PER04680
154       PTMIN(I) = 10000.0                                              PER04690
          VFEND(I) = 0.0                                                  PER04700
          DO 180 J = 1,20                                                 PER04710
             VF = FLOAT(J) * 10.0                                         PER04720
             CALL CONST                                                   PER04730
             ARMR = VF / (RVMR*RMR(I))                                    PER04740
             CALL VELMR                                                   PER04750
             CALL POWMR                                                   PER04760
             CALL VELTR                                                   PER04770
             CALL POWTR                                                   PER04780
             IF (PTAC(I) .GT. PTMIN(I)) GO TO 160                         PER04790
                PTMIN(I) = PTAC(I)                                        PER04800
```

```
              VFEND(I) = VF/1.68889
160        CONTINUE
180     CONTINUE
        VFEND(I) = VFEND(I) - 10.0
        VFF = VFEND(I)
        DO 220 J = 1,20
           VF = (FLOAT(J)-1.0) + VFF
           CALL CONST
           ARMR = VF / (RVMR*RMR(I))
           CALL VELMR
           CALL POWMR
           CALL VELTR
           CALL POWTR
           IF (PTAC(I) .GT. PTMIN(I)) GO TO 200
              PTMIN(I) = PTAC(I)
              VFEND(I) = VF/1.68889
200        CONTINUE
220     CONTINUE
        VFEND(I) = VFEND(I) - 1.0
        VFF = VFEND(I)
        DO 260 J = 1,200
           VF = ((FLOAT(J)/100)-0.01) + VFF
           CALL CONST
           ARMR = VF / (RVMR*RMR(I))
           CALL VELMR
           CALL POWMR
           CALL VELTR
           CALL POWTR
           IF (PTAC(I) .GT. PTMIN(I)) GO TO 240
              PTMIN(I) = PTAC(I)
              VFEND(I) = VF/1.68889
240        CONTINUE
260     CONTINUE
        VFEND(I) = VFEND(I) - 0.01
        IF(K .EQ. 1) VFEND1(I) = VFEND(I)
        IF(K .EQ. 2) VFEND2(I) = VFEND(I)
        IF(K .EQ. 3) VFEND3(I) = VFEND(I)
        IF(K .EQ. 4) VFEND4(I) = VFEND(I)
     IF(VFEND(I) .LT. MIN) MIN = VFEND(I)
     IF(VFEND(I) .GT. MAX) MAX = VFEND(I)
280  CONTINUE
C-----------------------------------------------------------------------
     MAXY = -500.0
     MINY = 500.0
     DO 480 I = 1,324
        DVFEND(I) = ((VFEND(I) - VFEND(163)) / VFEND(163)) * 100.0
        IF(DVFEND(I) .GT. MAXY) MAXY = DVFEND(I)
        IF(DVFEND(I) .LT. MINY) MINY = DVFEND(I)
```

```
480   CONTINUE                                                          PER05290
      IF(IANS .EQ. 3 .OR. IANS .EQ. 6) GO TO 482                        PER05300
      CALL GRAF2                                                        PER05310
482   CONTINUE                                                          PER05320
      RETURN                                                            PER05330
      END                                                               PER05340
C*******************************************************************    PER05350
C                                                                       PER05360
C       SUBROUTINE TO CALCULATE THE MAXIMUM RATE OF CLIMB               PER05370
C                                                                       PER05380
C*******************************************************************    PER05390
C*******************************************************************    PER05400
      SUBROUTINE MAXRC                                                  PER05410
      IMPLICIT REAL (A-H,L-Z)                                           PER05420
      COMMON /A/ AMR,ARMR,ARTR,ATR,CDCMR,CDOTR,CMR,CTMR,CTR,CTTR,       PER05430
     * DA,DEN,EFPAF,EFPAV,GE,HD,L,PA,RADIUS,RTR,RTHT,RVMR,RVTR,SDMR,    PER05440
     * SDTR,SKDHT,TEMP,W,VF,VV,IBTR,IBMR,CENSSL,PIE,PTMIN(400),VIMR,    PER05450
     * VITR,VTTR,VTMR,RANMAX,THETA,DELTA,SHPRAN(400),VV1,VF1,IENG,SDMR3,PER05460
     * VTMR2,DIFF,SFCM,SFCN,SFCC,SHPM,SHPN,SHPC,CMR1,RVMR1,RRMR         PER05470
       COMMON /B/ PIMR(400),POMR(400),PPMR(400),PCMR(400),PITR(400),    PER05480
     * PCTR(400),PTMR(400),PTTR(400),PTAC(400)                          PER05490
       COMMON /C/ RMR(400),I,J,K,DRMR(400),MINX,MAXX,MINY,MAXY,MINZ,    PER05500
     * MAXZ,DMIN,DMAX,MINB,MAXB,MINC,MAXC,MIN,MAX,MAXA,IANS             PER05510
      COMMON /W/ RCMAX(400),CRCMAX(400),VFRAN(400),DVFRAN(400),         PER05520
     * HOVALT(400),DHOVAL(400),SEVALT(400),DSEVAL(400),FMERT(400),      PER05530
     * DFMERT(400),VFEND(400),DVFEND(400)                               PER05540
       COMMON /Z/ VFEND1(400),VFEND2(400),VFEND3(400),VFEND4(400)       PER05550
C-------------------------------------------------------------------    PER05560
      PA = 0.0                                                          PER05570
      TEMP = 0.0                                                        PER05580
      DA = 0.0                                                          PER05590
      IF(IENG .EQ. 1) PTAVAL = (FLOAT(IENG)*SHPM - 10.0) * 0.97         PER05600
      IF(IENG .EQ. 2) PTAVAL = (FLOAT(IENG)*SHPM - 10.0)*0.9*0.97       PER05610
      IF(IENG .EQ. 3) PTAVAL = (FLOAT(IENG)*SHPM - 10.0)*0.9*0.94       PER05620
      CALL RHC                                                          PER05630
      MAX = 0.0                                                         PER05640
      MIN = 8000.0                                                      PER05650
      DO 580 I = 1,324                                                  PER05660
         AMR = RMR(I)**2.0*PIE                                          PER05670
         IF(K .NE.1) GO TO 490                                          PER05680
            VF = VFEND1(I)                                              PER05690
            VTMR = RVMR1*RMR(I)                                         PER05700
            SCMR = IBMR*CMR1 / (PIE*RMR(I))                             PER05710
490      IF(K .NE. 2) GO TO 492                                         PER05720
            VF = VFEND2(I)                                              PER05730
            RVMR = VTMR2/RMR(I)                                         PER05740
            VTMR = RVMR*RMR(I)                                          PER05750
            SCMR = IBMR*CMR1 / (PIE*RMR(I))                             PER05760
```

```
492       IF(K .NE. 3) GO TO 494
              VF = VFEND3(I)
              CMR = SDMR3*PIE*RMR(I)/IBMR
              SDMR =  IBMR*CMR / (PIE*RMR(I))
              VTMR =  RVMR1*RMR(I)
494       IF(K .NE. 4) GO TO 496
              VF = VFEND4(I)
              CMR = SDMR3*PIE*RMR(I)/IBMR
              SDMR =  IBMR*CMR / (PIE*RMR(I))
              RVMR =  VTMR2/RMR(I)
              VTMR =  RVMR*RMR(I)
496        DO 500 J = 1,80
              VV = FLOAT(J)*100.0
              CALL CONST
              ARMR =  VF /  (RVMR*RMR(I))
              CALL VELMR
              CALL POWMR
              CALL VELTR
              CALL POWTR
              IF (PTAC(I) .GT. PTAVAL) GO TO 520
              VF = VF/1.68889
500        CONTINUE
520        RCMAXX = (VV*60.0) - 100.0
           VF = VF/1.68889
           DO 540 J = 1,100
              VV = FLOAT(J)*1.0 + RCMAXX
              CALL CONST
              ARMR =  VF /  (RVMR*RMR(I))
              CALL VELMR
158           CALL POWMR
159           CALL VELTR
              CALL POWTR
              IF (PTAC(I) .GT. PTAVAL) GO TO 560
              VF = VF/1.68889
540        CONTINUE
560        RCMAX(I) = (VV*60.0)-1.0
      IF(RCMAX(I) .LT. MIN) MIN = RCMAX(I)
      IF(RCMAX(I) .GT. MAX) MAX = RCMAX(I)
580   CONTINUE
C-----------------------------------------------------------------------
      MAXY = -500.0
      MINY = 500.0
      DO 700 I = 1,324
           DRCMAX(I) = ((RCMAX(I) - RCMAX(163)) / RCMAX(163)) * 100.0
           IF(DRCMAX(I) .GT. MAXY) MAXY = DRCMAX(I)
           IF(DRCMAX(I) .LT. MINY) MINY = DRCMAX(I)
700   CONTINUE
      CALL GRAF3
```

```
      RETURN
      END
C*****************************************************************************
C
C     SUBROUTINE TO CALCULATE THE MAXIMUM RANGE VELOCITY
C
C*****************************************************************************
C*****************************************************************************
      SUBROUTINE RANGE
      IMPLICIT REAL (A-H,L-Z)
      COMMON /A/ AMR,ARMR,ARTR,ATR,CDOMR,CDOTR,CMR,CTMR,CTR,CTTR,
     * DA,DEN,EFPAF,EFPAV,GE,HD,L,PA,RADIUS,RTR,RTHT,RVMR,RVTR,SDMR,
     * SDTR,SKDHT,TEMP,W,VF,VV,IBTR,IBMR,DENSSL,PIE,PTMIN(400),VIMR,
     * VITR,VTTR,VTMR,RANMAX,THETA,DELTA,SHPRAN(400),VV1,VF1,IENG,SDMR3,
     * VTMR2,DIFF,SFCM,SFCN,SFCC,SHPM,SHPN,SHPC,CMR1,RVMR1,RRMR
      COMMON /B/ PIMR(400),POMR(400),PPMR(400),PCMR(400),PITR(400),
     * PCTR(400),PTMR(400),PTTR(400),PTAC(400)
      COMMON /C/ RMR(400),I,J,K,DRMR(400),MINX,MAXX,MINY,MAXY,MINZ,
     * MAXZ,DMIN,DMAX,MINB,MAXB,MINC,MAXC,MIN,MAX,MAXA,IANS,TEXT(200)
      COMMON /W/ RCMAX(400),DRCMAX(400),VFRAN(400),DVFRAN(400),
     * HOVALT(400),DHOVAL(400),SEVALT(400),DSEVAL(400),FMERT(400),
     * DFMERT(400),VFEND(400),DVFEND(400)
C-----------------------------------------------------------------------------
      MIN = 500.0
      MAX = -500.0
      PA = 4000.0
      TEMP = 95.0
      CALL RHO
      WFM = SFCM*SHPM
      WFN = SFCN*SHPN
      WFC = SFCC*SHPC
      BHAT = (WFM-WFC) / (SHPM-SHPC)
      AHAT = WFM - (BHAT*SHPM)
      PSHP = FLOAT(IENG)*AHAT*DELTA*SQRT(THETA)/BHAT
      DO 900 I = 1,324
         AMR = RMR(I)**2.0*PIE
         RANMAX = 0.0
         VV = 0.0
         IF(K .NE.1) GO TO 730
            VTMR = RVMR1*RMR(I)
            SDMR = IBMR*CMR1 / (PIE*RMR(I))
730      IF(K .NE. 2) GO TO 732
            RVMR = VTMR2/RMR(I)
            VTMR = RVMR*RMR(I)
            SDMR = IBMR*CMR1 / (PIE*RMR(I))
732      IF(K .NE. 3) GO TO 734
            CMR = SDMR3*PIE*RMR(I)/IBMR
            SDMR = IBMR*CMR / (PIE*RMR(I))
```

```
      VTMR = RVMR1*RMR(I)                                               PER06730
  734 IF(K .NE. 4) GO TO 736                                            PER06740
         CMR = SDMR3*PIE*RMR(I)/IBMR                                    PER06750
         SDMR = IBMR*CMR / (PIE*RMR(I))                                 PER06760
         RVMR = VTMR2/RMR(I)                                            PER06770
         VTMR = RVMR*RMR(I)                                             PER06780
  736 DO 740 J = 1,20                                                   PER06790
         VF = FLOAT(J)*10.0                                             PER06800
         CALL CONST                                                     PER06810
         ARMR = VF / (RVMR*RMR(I))                                      PER06820
         CALL VELMR                                                     PER06830
         CALL POWMR                                                     PER06840
         CALL VELTR                                                     PER06850
         CALL POWTR                                                     PER06860
         DE = 550.0*(PTAC(I)+PSHP) / VF                                 PER06870
         WDE = W / DE                                                   PER06880
         IF (WDE .LE. RANMAX) GO TO 750                                 PER06890
            RANMAX = WDE                                                PER06900
            VFRAN(I) = VF/1.68889                                       PER06910
  740 CONTINUE                                                          PER06920
  750 CONTINUE                                                          PER06930
      VFF = VFRAN(I)                                                    PER06940
      RANMAX = 0.0                                                      PER06950
      DO 820 J = 1,20                                                   PER06960
      VF = FLOAT(J)-1.0 + VFF                                           PER06970
         CALL CONST                                                     PER06980
         ARMR = VF / (RVMR*RMR(I))                                      PER06990
         CALL VELMR                                                     PER07000
         CALL POWMR                                                     PER07010
         CALL VELTR                                                     PER07020
         CALL POWTR                                                     PER07030
         DE = 550.0 * (PTAC(I)+PSHP) / VF                               PER07040
         IF(DE .EQ. 0.0) GO TO 780                                      PER07050
         WDE = W / DE                                                   PER07060
         GO TO 800                                                      PER07070
  780    WDE = 0.0                                                      PER07080
  800    IF (WDE .LT. RANMAX) GO TO 850                                 PER07090
            RANMAX = WDE                                                PER07100
            VFRAN(I) = VF/1.68889                                       PER07110
  820    CONTINUE                                                       PER07120
  850 CONTINUE                                                          PER07130
      VFF = VFRAN(I)                                                    PER07140
      RANMAX = 0.0                                                      PER07150
      DO 860 J = 1,201                                                  PER07160
      VF = (FLOAT(J-1)/100.0) + VFF                                     PER07170
         CALL CONST                                                     PER07180
         ARMR = VF / (RVMR*RMR(I))                                      PER07190
         CALL VELMR                                                     PER07200
```

```
            CALL POWMR
            CALL VELTR
            CALL POWTR
            DE = 550.0 * (PTAC(I)+PSHP) / VF
            WDE = W / DE
            IF (WDE .LT. RANMAX) GO TO 880
               RANMAX = WDE
               VFRAN(I) = VF/1.68889
               SHPRAN(I) = PTAC(I)
860         CONTINUE
880       CONTINUE
      IF(VFRAN(I) .LT. MIN) MIN = VFRAN(I)
      IF(VFRAN(I) .GT. MAX) MAX = VFRAN(I)
900   CONTINUE
C-----------------------------------------------------------------------
      MAXY = -500.0
      MINY = 500.0
      DO 1100 I = 1,324
         DVFRAN(I) = ((VFRAN(I) - VFRAN(163)) / VFRAN(163)) * 100.0
         IF(DVFRAN(I) .GT. MAXY) MAXY = DVFRAN(I)
         IF(DVFRAN(I) .LT. MINY) MINY = DVFRAN(I)
1100  CONTINUE
      IF ((MAXY - MINY) .GT. 0.5 ) GO TO 1102
      MAXY = MAXY + 1.5
      MINY = MINY - 1.5
      MIN = MIN - 0.50
      MAX = MAX + 0.50
1102  CONTINUE
      CALL GRAF4
      RETURN
      END
C***********************************************************************
C
C     SUBROUTINE TO CALCULATE THE MAXIMUM HOVER CEILING (IGE)
C
C***********************************************************************
C***********************************************************************
      SUBROUTINE HOVER
      IMPLICIT REAL (A-H,L-Z)
      COMMON /A/ AMR,ARMR,ARTR,ATR,CDOMR,CDOTR,CMR,CTMR,CTR,CTTR,
     * DA,DEN,EFPAF,EFPAV,GE,HD,L,PA,RADIUS,RTR,RTHT,RVMR,RVTR,SDMR,
     * SDTR,SKCHT,TEMP,W,VF,VV,IBTR,IBMR,DENSSL,PIE,PTMIN(400),VIMR,
     * VITR,VTTR,VTMR,RANMAX,THETA,DELTA,SHPRAN(400),VV1,VF1,IENG,SDMR3,
     * VTMR2,DIFF,SFCM,SFCN,SFCC,SHPM,SHPN,SHPC,CMR1,RVMR1,RRMR
      COMMON /B/ PIMR(400),POMR(400),PPMR(400),PCMR(400),PITR(400),
     * POTR(400),PTMR(400),PTTR(400),PTAC(400)
      COMMON /C/ RMR(400),I,J,K,DRMR(400),MINX,MAXX,MINY,MAXY,MINZ,
     * MAXZ,DMIN,DMAX,MINB,MAXB,MINC,MAXC,MIN,MAX,MAXA,IANS,TEXT(200)
```

```
      COMMON /W/ RCMAX(400),DRCMAX(400),VFRAN(400),DVFRAN(400),
     * HCVALT(400),DHOVAL(400),SEVALT(400),DSEVAL(400),FMERT(400),
     * DFMERT(400),VFEND(400),DVFEND(400)
C-----------------------------------------------------------------------
      VF = 0.C
      VV = 0.C
      SKCHT = 10.0
      MIN = 3C000.0
      MAX = 0.0
      PA = 0.0
      TEMP = C.0
      IF(IENG .EQ. 1) PTAVAL = (FLOAT(IENG)*SHPN - 10.0) * 0.97
      IF(IENG .EQ. 2) PTAVAL = (FLOAT(IENG)*SHPN - 10.0)*0.9*C.97
      IF(IENG .EQ. 3) PTAVAL = (FLOAT(IENG)*SHPN - 10.0)*0.9*0.94
      DO 1240 I = 1,324
         AMR = RMR(I)**2.0*PIE
         IF(K .NE.1) GO TO 1108
            VTMR = RVMR1*RMR(I)
            SDMR = IBMR*CMR1 / (PIE*RMR(I))
1108     IF(K .NE. 2) GO TO 1112
            RVMR = VTMR2/RMR(I)
            VTMR = RVMR*RMR(I)
            SCMR = IBMR*CMR1 / (PIE*RMR(I))
1112     IF(K .NE. 3) GO TO 1114
            CMR = SDMR3*PIE*RMR(I)/IBMR
            SCMR = IBMR*CMR / (PIE*RMR(I))
            VTMR = RVMR1*RMR(I)
1114     IF(K .NE. 4) GO TO 1116
            CMR = SDMR3*PIE*RMR(I)/IBMR
            SDMR = IBMR*CMR / (PIE*RMR(I))
            RVMR = VTMR2/RMR(I)
            VTMR = RVMR*RMR(I)
1116     ARMR = VF / (RVMR*RMR(I))
         DO 1120 J = 1,30
            CA = FLOAT(J)*1000.0
            CALL RHO
            CALL CONST
            CALL VELMR
            CALL POWMR
            CALL VELTR
            CALL POWTR
            POWAA = PTAVAL*DELTA*SQRT(THETA)
            IF (PTAC(I) .LT. POWAA) GOTC 1120
               HCVALT(I) = CA
               GC TO 1140
1120        CCNTINUE
1140     CONTINUE
         HOV = HOVALT(I) - 1000.0
```

```
      DO 1160 J = 1,20
           DA = FLOAT(J)*50.0 + HOV
           CALL RHO
           CALL CONST
           CALL VELMR
           CALL POWMR
           CALL VELTR
           CALL POWTR
           POWAA = PTAVAL*DELTA*SQRT(THETA)
           IF (PTAC(I) .LT. POWAA) GOTO 1160
              HOVALT(I) = DA
              GO TO 1180
1160       CONTINUE
1180  CONTINUE
      HOV = HOVALT(I) - 50.0
      DO 1200 J = 1,100
           DA = FLOAT(J)*0.5 + HOV
           CALL RHO
           CALL CONST
           CALL VELMR
           CALL POWMR
           CALL VELTR
           CALL POWTR
           POWAA = PTAVAL*DELTA*SQRT(THETA)
           IF (PTAC(I) .LT. POWAA) GOTO 1200
              HOVALT(I) = DA-0.5
              GO TO 1220
1200       CONTINUE
1220  CONTINUE
   IF(HOVALT(I) .LT. MIN) MIN = HOVALT(I)
   IF(HOVALT(I) .GT. MAX) MAX = HOVALT(I)
1240  CONTINUE
C---------------------------------------------------------------------
      MAXY = -500.0
      MINY = 500.0
      DO 1400 I = 1,324
          DHOVAL(I) = ((HOVALT(I) - HOVALT(163)) / HOVALT(163)) * 100.0
          IF(DHOVAL(I) .GT. MAXY) MAXY = DHOVAL(I)
          IF(DHOVAL(I) .LT. MINY) MINY = DHOVAL(I)
1400  CONTINUE
      CALL GRAF5
      RETURN
      END
C*********************************************************************
C
C     SUBROUTINE TO CALCULATE THE MAXIMUM SERVICE CEILING
C
C*********************************************************************
```

```
C*****************************************************************************
      SUBROUTINE CEIL
      IMPLICIT REAL (A-H,L-Z)
      COMMON /A/ AMR,ARMR,ARTR,ATR,CDCMR,CDOTR,CMR,CTMR,CTR,CTTR,
     * DA,DEN,EFPAF,EFPAV,GE,HD,L,PA,RADIUS,RTR,RTHT,RVMR,RVTR,SDMR,
     * SDTR,SKCHT,TEMP,W,VF,VV,IBTR,IBMR,DENSSL,PIE,PTMIN(400),VIMR,
     * VITR,VTTR,VTMR,RANMAX,THETA,DELTA,SHPRAN(400),VV1,VF1,IENG,SDMR3,
     * VTMR2,DIFF,SFCM,SFCN,SFCC,SHPM,SHPN,SHFC,CMR1,RVMR1,RRMR
      COMMON /B/ PIMR(400),POMR(400),PPMR(400),PCMR(400),PITR(400),
     * POTR(400),PTMR(400),PTTR(400),PTAC(400)
      COMMON /C/ RMR(400),I,J,K,DRMR(400),MINX,MAXX,MINY,MAXY,MINZ,
     * MAXZ,DMIN,DMAX,MINB,MAXB,MINC,MAXC,MIN,MAX,MAXA,IANS,TEXT(200)
      COMMON /W/ RCMAX(400),DRCMAX(400),VFRAN(400),DVFRAN(400),
     * HOVALT(400),DHOVAL(400),SEVALT(400),DSEVAL(400),FMERT(400),
     * DFMERT(400),VFEND(400),DVFEND(400)
      COMMON /Z/ VFEND1(400),VFEND2(400),VFEND3(400),VFEND4(400)
C-----------------------------------------------------------------------------
      SKCHT = 2500.0
      MIN = 30000.0
      MAX = 0.0
      PA = 0.0
      TEMP = 0.0
      IF(IENG .EQ. 1) PTAVAL = (FLOAT(IENG)*SHPN - 10.0) * 0.97
      IF(IENG .EQ. 2) PTAVAL = (FLOAT(IENG)*SHPN - 10.0)*0.9*0.97
      IF(IENG .EQ. 3) PTAVAL = (FLOAT(IENG)*SHPN - 10.0)*0.9*0.94
      DO 1540 I = 1,324
         AMR = RMR(I)**2.0*PIE
         IF(K .NE.1) GO TO 1408
            VFF = VFEND1(I)
            VTMR = RVMR1*RMR(I)
            SDMR = IBMR*CMR1 / (PIE*RMR(I))
1408     IF(K .NE. 2) GO TO 1412
            VFF = VFEND2(I)
            RVMR = VTMR2/RMR(I)
            VTMR = RVMR*RMR(I)
            SDMR = IBMR*CMR1 / (PIE*RMR(I))
1412     IF(K .NE. 3) GO TO 1414
            VFF = VFEND3(I)
            CMR = SDMR3*PIE*RMR(I)/IBMR
            SDMR = IBMR*CMR / (PIE*RMR(I))
            VTMR = RVMR1*RMR(I)
1414     IF(K .NE. 4) GO TO 1416
            CMR = SDMR3*PIE*RMR(I)/IBMR
            SDMR = IBMR*CMR / (PIE*RMR(I))
            RVMR = VTMR2/RMR(I)
            VTMR = RVMR*RMR(I)
            VFF = VFEND4(I)
1416     DO 1420 J = 1,30
```

```
      VF = VFF
      DA = FLOAT(J)*1000.0
      CALL RHO
      VV = 100.0
      CALL CONST
      ARMR = VF / (RVMR*RMR(I))
      CALL VELMR
      CALL POWMR
      CALL VELTR
      CALL POWTR
      FCWAA = PTAVAL*DELTA*SQRT(THETA)
      IF (PTAC(I) .LT. POWAA) GO TO 1420
         SEVALT(I) = DA
         GO TO 1440
1420    CONTINUE
1440  CONTINUE
      SEV = SEVALT(I) - 1000.0
      DO 1460 J = 1,20
      VF = VFF
      DA = FLOAT(J)*50.0 + SEV
      CALL RHO
      VV = 100.0
      CALL CONST
      ARMR = VF / (RVMR*RMR(I))
      CALL VELMR
      CALL POWMR
      CALL VELTR
      CALL POWTR
      FCWAA = PTAVAL*DELTA*SQRT(THETA)
      IF (PTAC(I) .LT. POWAA) GO TO 1460
         SEVALT(I) = DA
         GO TO 1480
1460    CONTINUE
1480  CONTINUE
      SEV = SEVALT(I) - 50.0
      DO 1500 J = 1,100
      VF = VFF
      DA = FLOAT(J)*0.5 + SEV
      CALL RHO
      VV = 100.0
      CALL CONST
      ARMR = VF / (RVMR*RMR(I))
      CALL VELMR
      CALL POWMR
      CALL VELTR
      CALL POWTR
      FCWAA = PTAVAL*DELTA*SQRT(THETA)
      IF (PTAC(I) .LT. POWAA) GO TO 1500
```

```
              SEVALT(I) = DA
              GO TO 1520
1500        CONTINUE
1520     CONTINUE
         SEVALT(I) = SEVALT(I) - 0.5
      IF(SEVALT(I) .LT. MIN) MIN = SEVALT(I)
      IF(SEVALT(I) .GT. MAX) MAX = SEVALT(I)
1540  CONTINUE
C-----------------------------------------------------------------------
      MAXY = -500.0
      MINY = 500.0
      DO 1700 I = 1,324
         DSEVAL(I) = ((SEVALT(I) - SEVALT(163)) / SEVALT(163)) * 100.0
         DRMR(I) = ((RMR(I) - RMR(163)) / RMR(163)) * 100.0
         IF(DSEVAL(I) .GT. MAXY) MAXY = DSEVAL(I)
         IF(DSEVAL(I) .LT. MINY) MINY = DSEVAL(I)
1700  CONTINUE
      CALL GRAF6
      RETURN
      END
C=======================================================================
C==============                                   ======================
C=======================    SUB-GROUP A    ============================
C==============    GENERAL USE SUBROUTINES    =========================
C================                          ============================
C=======================================================================
C***********************************************************************
C
C     SUBROUTINE TO CALCULATE DENSITY ALTITUDE
C
C***********************************************************************
C***********************************************************************
      SUBROUTINE RHO
      IMPLICIT REAL (A-H,L-Z)
      COMMON /A/ AMR,ARMR,ARTR,ATR,CDCMR,CDOTR,CMR,CTMR,CTR,CTTR,
     * DA,DEN,EFPAF,EFPAV,GE,HD,L,PA,RADIUS,RTR,RTHT,RVMR,RVTR,SDMR,
     * SDTR,SKDHT,TEMP,W,VF,VV,IBTR,IBMR,DENSSL,PIE,PTMIN(400),VIMR,
     * VITR,VTTR,VTMR,RANMAX,THETA,DELTA,SHPRAN(400),VV1,VF1,IENG,SDMR3,
     * VTMR2,DIFF,SFCM,SFCN,SFCC,SHPM,SHPN,SHPC,CMR1,RVMR1
C-----------------------------------------------------------------------
      IF (TEMP .EQ. 0.0 .OR. PA .EQ. 0.0 ) GO TO 10
      H = (1.0-(PA*6.875E-06))**5.2561 / ((TEMP-32.0)*0.5555+273.16)
      DA = (1.0-(H*288.16)**0.23496) / 6.875E-06
10    DEN = 0.0023769*(1.0-6.875E-06*DA)**4.2561
      TEMP1 = 518.688*(1.0-6.875E-06*DA)
      PRESS1 = 2116.22*(1.0-6.875E-06*DA)**5.2561
      DEN1 = DENSSL*(1.0-6.875E-06*DA)**4.2561
      THETA = TEMP1/518.688
```

```
      DELTA = PRESS1/2116.22
      RETURN
      END
C***********************************************************************
C
C    SUBROUTINE TO CALCULATE INDUCED VELOCITY FOR MAIN ROTOR
C
C***********************************************************************
C***********************************************************************
      SUBROUTINE VELMR
      IMPLICIT REAL (A-H,L-Z)
      COMMON /A/ AMR,ARMR,ARTR,ATR,CDCMR,CDOTR,CMR,CTMR,CTR,CTTR,
     * DA,DEN,EFPAF,EFPAV,GE,HD,L,PA,RADIUS,RTR,RTHT,RVMR,RVTR,SDMR,
     * SDTR,SKDHT,TEMP,W,VF,VV,IBTR,IBMR,DENSSL,PIE,PTMIN(400),VIMR,
     * VITR,VTTR,VTMR,RANMAX,THETA,DELTA,SHPRAN(400),VV1,VF1,IENG,SDMR3,
     * VTMR2,DIFF,SFCM,SFCN,SFCC,SHPM,SHPN,SHPC,CMR1,RVMR1
C-----------------------------------------------------------------------
      VIHMR = SQRT(W/(2.0*DEN*AMR))
      VITMR = SQRT(-(VF**2./2.0) + SQRT((VF**2./2.0)**2.0 + VIHMR**4))
      VIMR = VITMR
      IF (VF .NE. 0.0 .OR. VV .EQ. 0.0) GO TO 10
      VIVMR = 0.5*(-VV + SQRT(VV + 4*VIHMR))
      VIMR = VIVMR
10    CONTINUE
      RETURN
      END
C***********************************************************************
C
C     SUBROUTINE TO CALCULATE INDUCED VELOCITY FOR TAIL ROTOR
C
C***********************************************************************
C***********************************************************************
      SUBROUTINE VELTR
      IMPLICIT REAL (A-H,L-Z)
      COMMON /A/ AMR,ARMR,ARTR,ATR,CDCMR,CDOTR,CMR,CTMR,CTR,CTTR,
     * DA,DEN,EFPAF,EFPAV,GE,HD,L,PA,RADIUS,RTR,RTHT,RVMR,RVTR,SDMR,
     * SDTR,SKDHT,TEMP,W,VF,VV,IBTR,IBMR,DENSSL,PIE,PTMIN(400),VIMR,
     * VITR,VTTR,VTMR,RANMAX,THETA,DELTA,SHPRAN(400),VV1,VF1,IENG,SDMR3,
     * VTMR2,DIFF,SFCM,SFCN,SFCC,SHPM,SHPN,SHPC,CMR1,RVMR1
      COMMON /C/ RMR(400),I,J,K,DRMR(400),MINX,MAXX,MINY,MAXY,MINZ,
     * MAXZ,CMIN,CMAX,MINB,MAXB,MINC,MAXC,MIN,MAX,MAXA,IANS
      COMMON /B/ PIMR(400),POMR(400),PPMR(400),PCMR(400),PITR(400),
     * POTR(400),PTMR(400),PTTR(400),PTAC(400)
C-----------------------------------------------------------------------
      VIHTR = SQRT(PTMR(I)*550.0/(2.0*DEN*ATR*RVMR*L))
      VITTR = SQRT(-(VF**2./2.)+SQRT((VF**2./2.)**2.+VIHTR**4.))
      VITR = VITTR
      RETURN
```

```
      END                                                               PER10570
C***********************************************************************PER10580
C                                                                       PER10590
C     SUBROUTINE TO CALCULATE POWER REQUIREMENTS FOR MAIN ROTOR         PER10600
C                                                                       PER10610
C***********************************************************************PER10620
C***********************************************************************PER10630
      SUBROUTINE POWMR                                                  PER10640
      IMPLICIT REAL (A-H,L-Z)                                           PER10650
      COMMON /A/ AMR,ARMR,ARTR,ATR,CDOMR,CDOTR,CMR,CTMR,CTR,CTTR,       PER10660
     * DA,DEN,EFPAF,EFPAV,GE,HD,L,PA,RADIUS,RTR,RTHT,RVMR,RVTR,SDMR,    PER10670
     * SDTR,SKCHT,TEMP,W,VF,VV,IBTR,IBMR,DENSSL,PIE,PTMIN(400),VIMR,    PER10680
     * VITR,VTTR,VTMR,RANMAX,THETA,DELTA,SHPRAN(400),VV1,VF1,IENG,SDMR3,PER10690
     * VTMR2,DIFF,SFCM,SFCN,SFCC,SHPM,SHPN,SHPC,CMR1,RVMR1              PER10700
      COMMON /B/ PIMR(400),POMR(400),PPMR(400),PCMR(400),PITR(400),     PER10710
     * POTR(400),PTMR(400),PTTR(400),PTAC(400)                          PER10720
      COMMON /C/ RMR(400),I,J,K,DRMR(400),MINX,MAXX,MINY,MAXY,MINZ,     PER10730
     * MAXZ,DMIN,DMAX,MINB,MAXB,MINC,MAXC,MIN,MAX,MAXA,IANS             PER10740
C-----------------------------------------------------------------------PER10750
      CTMR = W/(DEN*AMR*VTMR**2.0)                                      PER10760
      TLMR = 1.0 - (SQRT(2.0*CTMR) / FLOAT(IBMR))                       PER10770
      PIMR(I) = W*VIMR*GE / (TLMR*550.0)                                PER10780
      POMR(I) = (SDMR*CDOMR*DEN*AMR*VTMR**3.0) * (1.0 + 4.3*ARMR**2.0)/ PER10790
     * (550.0 * 8.0)                                                    PER10800
      PPMR(I) = 0.5*DEN * (VF**3.0*EFPAF + VV**3.0*EFPAV) / 550.0       PER10810
      PCMR(I) = W*VV / 550.0                                            PER10820
      PTMR(I) = PIMR(I) + PCMR(I) + PPMR(I) + PCMR(I)                   PER10830
      RETURN                                                            PER10840
      END                                                               PER10850
C***********************************************************************PER10860
C                                                                       PER10870
C     SUBROUTINE TO CALCULATE POWER REQUIREMENTS FOR TAIL ROTOR         PER10880
C         AND POWER FOR TOTAL AIRCRAFT.                                 PER10890
C                                                                       PER10900
C***********************************************************************PER10910
C***********************************************************************PER10920
      SUBROUTINE POWTR                                                  PER10930
      IMPLICIT REAL (A-H,L-Z)                                           PER10940
      COMMON /A/ AMR,ARMR,ARTR,ATR,CDOMR,CDOTR,CMR,CTMR,CTR,CTTR,       PER10950
     * DA,DEN,EFPAF,EFPAV,GE,HD,L,PA,RADIUS,RTR,RTHT,RVMR,RVTR,SDMR,    PER10960
     * SDTR,SKCHT,TEMP,W,VF,VV,IBTR,IBMR,DENSSL,PIE,PTMIN(400),VIMR,    PER10970
     * VITR,VTTR,VTMR,RANMAX,THETA,DELTA,SHPRAN(400),VV1,VF1,IENG,SDMR3,PER10980
     * VTMR2,DIFF,SFCM,SFCN,SFCC,SHPM,SHPN,SHPC,CMR1,RVMR1              PER10990
      COMMON /B/ PIMR(400),POMR(400),PPMR(400),PCMR(400),PITR(400),     PER11000
     * POTR(400),PTMR(400),PTTR(400),PTAC(400)                          PER11010
      COMMON /C/ RMR(400),I,J,K,DRMR(400),MINX,MAXX,MINY,MAXY,MINZ,     PER11020
     * MAXZ,DMIN,DMAX,MINB,MAXB,MINC,MAXC,MIN,MAX,MAXA,IANS             PER11030
C-----------------------------------------------------------------------PER11040
```

```
      CTTR = PTMR(I)*550.0 / (DEN*ATR*RVMR*L*VTTR**2.0)
      TLTR = 1.0 - SQRT(2.0*CTTR) / IBTR
      PITR(I) = PTMR(I)*VITR / (L*RVMR*TLTR)
      POTR(I) = (SDTR*CDOTR*ATR*DEN*VTTR**3.0)* (1.0 + 4.3*ARTR**2.0)/
     * (550.0 * 8.0)
      PTTR(I) = PITR(I) + PCTR(I)
      PTAC(I) = PTMR(I) + PTTR(I)
      RETURN
      END
C*****************************************************************
C
C     SUBROUTINE TO CALCULATE CONSTANTS
C
C*****************************************************************
C*****************************************************************
      SUBROUTINE CONST
      IMPLICIT REAL (A-H,L-Z)
      COMMON /A/ AMR,ARMR,ARTR,ATR,CDCMR,CDOTR,CMR,CTMR,CTR,CTTR,
     * DA,DEN,EFPAF,EFPAV,GE,HD,L,PA,RADIUS,RTR,RTHT,RVMR,RVTR,SDMR,
     * SDTR,SKDHT,TEMP,W,VF,VV,IBTR,IBMR,DENSSL,PIE,PTMIN(400),VIMR,
     * VITR,VTTR,VTMR,RANMAX,THETA,DELTA,SHPRAN(400),VV1,VF1,IENG,SDMR3,
     * VTMR2,DIFF,SFCM,SFCN,SFCC,SHPM,SHPN,SHPC,CMR1,RVMR1
      COMMON /C/ RMR(400),I,J,K,DRMR(400),MINX,MAXX,MINY,MAXY,MINZ,
     * MAXZ,DMIN,DMAX,MINB,MAXB,MINC,MAXC,MIN,MAX,MAXA,IANS
C-----------------------------------------------------------------
      PIE = 3.141593
      VF = VF * 1.68889
      VV = VV / 60.0
      ATR = PIE * RTR**2.0
      VTTR = RVTR * RTR
      ARTR = VF / (RVTR*RTR)
      SDTR = IBTR*CTR / (PIE*RTR)
      HD = (SKDHT+RTHT) / (RMR(I)*2.0)
      IF(HD .GT. 1.6) GO TO 10
      GE = -0.1276*HD**4 + 0.708*HD**3 - 1.4569*HD**2 + 1.3432*HD +
     *  0.5147
      GO TO 11
10    GE = 1.0
11    CONTINUE
      RETURN
      END
C=================================================================
C=============                          =========================
C===================      SUB-GROUP C     =========================
C=============          GRAPHS SUBROUTINES   ======================
C===============                        ==========================
C=================================================================
C*****************************************************************
```

```
C
C     SUBROUTINE TO CREATE GRAPHS FOR POWER AND FIGURE OF MERIT
C         VS RADIUS
C
C*********************************************************************
C*********************************************************************
      SUBROUTINE GRAF1C
      IMPLICIT REAL (A-H,L-Z)
       COMMON /B/ PIMR(400),POMR(400),PPMR(400),PCMR(400),PITR(400),
     * POTR(400),PTMR(400),PTTR(400),PTAC(400)
       COMMON /C/ RMR(400),I,J,K,DRMR(400),MINX,MAXX,MINY,MAXY,MINZ,
     * MAXZ,DMIN,DMAX,MINB,MAXB,MINC,MAXC,MIN,MAX,MAXA,IANS,TEXT(200)
       COMMON /Y/DPIMR(400),DPOMR(400),DPPMR(400),DPCMR(400),DPTMR(400),
     * DPITR(400),DPOTR(400),DPTAC(400),DPTTR(400)
      CALL COMPRS
      CALL PAGE (11.0,8.5)
      CALL PHYSOR(1.5,1.5)
      CALL AREA2D (2.50,4.50)
      CALL XNAME ('RADIUS OF MAIN ROTOR (FT)$',100)
      CALL YNAME ('POWER REQUIRED (SHP)$',100)
      CALL GRAF (MINZ,'SCALE',MAXZ,0.0,'SCALE',MAX)
      CALL LEGLIN
      CALL DASH
      CALL CURVE (RMR,PTTR,322,0)
      CALL CHNDOT
      CALL CURVE (RMR,PTMR,322,0)
      CALL CHNDSH
      CALL CURVE (RMR,PTAC,322,0)
      MAXLIN=LINEST(TEXT,100,30)
      CALL HEIGHT(0.1)
      CALL LINES('TAIL ROTOR POWER$',TEXT,1)
      CALL LINES('MAIN ROTOR POWER$',TEXT,2)
      CALL LINES('TOTAL POWER$',TEXT,3)
      CALL LEGEND(TEXT,3,0.4,1.5)
      CALL BLREC(0.3,1.42,1.95,0.75,1.2)
      CALL RESET('HEIGHT')
      CALL DOT
      CALL GRID (1,2)
      CALL ENDGR(0)
C---------------------------------------------------------------------
      CALL PHYSOR(5.5,1.5)
      CALL AREA2D (2.50,4.50)
      CALL XNAME ('RADIUS OF MAIN ROTOR (% CHANGE)$',100)
      CALL YNAME ('POWER REQUIRED (% CHANGE)$',100)
      CALL GRAF (DMIN,'SCALE',DMAX,MINY,'SCALE',MAXY)
      CALL LEGLIN
      CALL DASH
      CALL CURVE (DRMR,DPTTR,324,0)
```

```
      CALL CHNDOT
      CALL CURVE (DRMR,DPTMR,322,0)
      CALL CHNDSH
      CALL CURVE (DRMR,DPTAC,322,0)
      MAXLIN=LINEST(TEXT,100,30)
      CALL HEIGHT(0.1)
      CALL LINES('TAIL ROTOR POWER$',TEXT,1)
      CALL LINES('MAIN ROTOR POWER$',TEXT,2)
      CALL LINES('TOTAL POWER$',TEXT,3)
      CALL LEGEND(TEXT,3,0.4,3.8)
      CALL BLREC(0.3,3.7,1.95,0.75,1.2)
      CALL RESET('HEIGHT')
      CALL DOT
      CALL GRID (1,2)
      CALL ENDGR(0)
C-----------------------------------------------------------------------
      CALL PHYSOR(1.5,1.5)
      CALL AREA2D (6.50,4.70)
      CALL HEADIN ('POWER VERSUS RADIUS$',19,1.4,3)
      IF(K .NE. 1 ) GO TO 8
      CALL HEADIN ('CHORD & ROTATIONAL VELOCITY HELD CONSTANT$',41,
     *            0.8,3)
      CALL HEADIN ('SOLIDITY, TIP VELOCITY & ADVANCE RATIO$',38,0.8,4)
      CALL HEADIN('ALLOWED TO VARY WITH RADIUS$',27,0.8,4)
      CALL MESSAG ('SPEC RMR =$',100,7.4,6.5)
      CALL REALNO (RMR(163),2,'ABUT','ABUT')
      CALL MESSAG('CASE 1$',100,8.7,6.0)
8     IF(K .NE. 2) GO TO 10
      CALL HEADIN ('CHORD,TIP VELOCITY & ADVANCE RATIO HELD CONSTANT$',
     *      48,0.8,4)
      CALL HEADIN ('SOLIDITY, ROTATIONAL VELOCITY$',29,0.8,4)
      CALL HEADIN ('ALLOWED TO VARY WITH RADIUS$',27,0.8,4)
      CALL MESSAG ('SPEC RMR =$',100,7.4,6.5)
      CALL REALNO (RMR(163),2,'ABUT','ABUT')
      CALL MESSAG('CASE 2$',100,8.7,6.0)
10    IF( K .NE. 3) GO TO 12
      CALL HEADIN ('SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT$',
     *      44,0.8,4)
      CALL HEADIN ('CHORD,ADVANCE RATIO & TIP VELOCITY$',34,0.8,4)
      CALL HEADIN ('ALLOWED TO VARY WITH RADIUS$',27,0.8,4)
      CALL MESSAG ('SPEC RMR =$',100,7.4,6.5)
      CALL REALNO (RMR(163),2,'ABUT','ABUT')
      CALL MESSAG('CASE 3$',100,8.7,6.0)
12    IF( K .NE. 4) GO TO 14
      CALL HEADIN('SOLIDITY,ADVANCE RATIO & TIP VELOCITY HELD CONSTANT$'
     *,51,0.8,4)
      CALL HEADIN ('CHORD & ROTATIONAL VELOCITY$',27,0.8,4)
      CALL HEADIN ('ALLOWED TO VARY WITH RADIUS$',27,0.8,4)
```

```
      CALL MESSAG ('SPEC RMR =$',100,7.4,6.5)                              PER12490
      CALL REALNO (RMR(163),2,'ABUT','ABUT')                               PER12500
      CALL MESSAG('CASE 4$',100,8.7,6.0)                                   PER12510
14    CALL ENDPL(0)                                                        PER12520
      RETURN                                                               PER12530
      END                                                                  PER12540
C******************************************************************        PER12550
C                                                                          PER12560
C     SUBROUTINE  TO CREATE GRAPH OF POWER (INDUCED,PROFILE, ETC.)         PER12570
C          VS RADIUS.                                                      PER12580
C                                                                          PER12590
C******************************************************************        PER12600
C******************************************************************        PER12610
      SUBROUTINE GRAF1D                                                    PER12620
      IMPLICIT REAL (A-H,L-Z)                                              PER12630
       COMMON /B/ PIMR(400),POMR(400),PPMR(400),PCMR(400),PITR(400),       PER12640
     * POTR(400),PTMR(400),PTTR(400),PTAC(400)                             PER12650
       COMMON /C/ RMR(400),I,J,K,DRMR(400),MINX,MAXX,MINY,MAXY,MINZ,       PER12660
     * MAXZ,DMIN,DMAX,MINB,MAXB,MINC,MAXC,MIN,MAX,MAXA,IANS,TEXT(200)      PER12670
       COMMON /Y/DPIMR(400),DPOMR(400),DPPMR(400),DPCMR(400),DPTMR(400),   PER12680
     * DPITR(400),DPOTR(400),DPTAC(400)                                    PER12690
      CALL COMPRS                                                          PER12700
      CALL PAGE (11.0,8.5)                                                 PER12710
      CALL PHYSOR(1.5,1.5)                                                 PER12720
      CALL AREA2D (2.50,4.50)                                              PER12730
      CALL XNAME ('RADIUS OF MAIN ROTOR (FT)$',100)                        PER12740
      CALL YNAME ('POWER REQUIRED (SHP)$',100)                             PER12750
      CALL GRAF (MINZ,'SCALE',MAXZ,0.0,'SCALE',MAXA)                       PER12760
      CALL LEGLIN                                                          PER12770
      CALL CURVE (RMR,PIMR,324,0)                                          PER12780
      CALL DASH                                                            PER12790
      CALL CURVE (RMR,POMR,324,0)                                          PER12800
      CALL CHNDOT                                                          PER12810
      CALL CURVE (RMR,PPMR,324,0)                                          PER12820
      CALL CHNDSH                                                          PER12830
      CALL CURVE (RMR,PITR,324,0)                                          PER12840
      CALL DOT                                                             PER12850
      CALL CURVE (RMR,POTR,324,0)                                          PER12860
      IF (VV .LE. 0.0) GO TO 20                                            PER12870
      CALL CURVE (RMR,PCMR,324,0)                                          PER12880
20    MAXLIN=LINEST(TEXT,100,30)                                           PER12890
      CALL HEIGHT(0.1)                                                     PER12900
      CALL LINES('INDUCED POWER (MR)$',TEXT,1)                             PER12910
      CALL LINES('PROFILE POWER (MR)$',TEXT,2)                             PER12920
      CALL LINES('PARASITE POWER (MR)$',TEXT,3)                            PER12930
      CALL LINES('INDUCED POWER (TR)$',TEXT,4)                             PER12940
      CALL LINES('PROFILE POWER (TR)$',TEXT,5)                             PER12950
      CALL LEGEND(TEXT,5,0.4,3.50)                                         PER12960
```

```
      CALL BLREC(0.3,3.42,2.15,1.00,1.1)
      CALL RESET('HEIGHT')
      CALL DOT
      CALL GRID (1,2)
      CALL ENDGR(0)
C-----------------------------------------------------------------------
      CALL PHYSOR(5.5,1.5)
      CALL AREA2D (2.50,4.50)
      CALL XNAME ('RADIUS OF MAIN ROTOR (% CHANGE)$',100)
      CALL YNAME ('POWER REQUIRED (% CHANGE)$',100)
      CALL GRAF (DMIN,'SCALE',DMAX,MINX,'SCALE',MAXX)
      CALL LEGLIN
      CALL CURVE (DRMR,DPIMR,324,0)
      CALL DASH
      CALL CURVE (DRMR,DPOMR,324,0)
      CALL CHNDOT
      CALL CURVE (DRMR,DPPMR,324,0)
      CALL CHNDSH
      CALL CURVE (DRMR,DPITR,324,0)
      CALL DOT
      CALL CURVE (DRMR,DPOTR,324,0)
      IF (VV .LE. 0.0) GO TO 22
      CALL CURVE (DRMR,DPCMR,322,0)
22    MAXLIN=LINEST(TEXT,100,30)
      CALL HEIGHT(0.1)
      CALL LINES('INDUCED POWER (MR)$',TEXT,1)
      CALL LINES('PROFILE POWER (MR)$',TEXT,2)
      CALL LINES('PARASITE POWER (MR)$',TEXT,3)
      CALL LINES('INDUCED POWER (TR)$',TEXT,4)
      CALL LINES('PROFILE POWER (TR)$',TEXT,5)
      CALL LEGEND(TEXT,5,0.4,3.50)
      CALL BLREC(0.3,3.42,2.2,1.00,1.2)
      CALL RESET('HEIGHT')
      CALL DOT
      CALL GRID (1,2)
      CALL ENDGR(0)
C-----------------------------------------------------------------------
      CALL PHYSOR(1.5,1.5)
      CALL AREA2D (6.50,4.70)
      CALL HEADIN ('POWER VERSUS RADIUS$',19,1.4,3)
      IF(K .NE. 1 ) GO TO 8
      CALL HEADIN ('CHORD & ROTATIONAL VELOCITY HELD CONSTANT$',41,
     *            0.8,3)
      CALL HEADIN ('SOLIDITY, TIP VELOCITY & ADVANCE RATIO$',38,0.8,4)
      CALL HEADIN('ALLOWED TO VARY WITH RADIUS$',27,0.8, 4)
      CALL MESSAG ('SPEC RMR =$',100,7.4,6.5)
      CALL REALNO (RMR(163),2,'ABUT','ABUT')
      CALL MESSAG('CASE 1$',100,8.7,6.0)
```

```
8     IF(K .NE. 2) GO TO 10
      CALL HEADIN ('CHORD,TIP VELOCITY & ADVANCE RATIO HELD CONSTANT$',
     *     48,0.8,4)
      CALL HEADIN ('SOLIDITY, ROTATIONAL VELOCITY$',29,0.8,4)
      CALL HEADIN ('ALLOWED TO VARY WITH RADIUS$',27,0.8,4)
      CALL MESSAG ('SPEC RMR =$',100,7.4,6.5)
      CALL REALNO (RMR(163),2,'ABUT','ABUT')
      CALL MESSAG('CASE 2$',100,8.7,6.0)
10    IF( K .NE. 3) GO TO 12
      CALL HEADIN (' SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT$',
     *     44,0.8,4)
      CALL HEADIN ('CHORD,ADVANCE RATIO & TIP VELOCITY$',34,0.8,4)
      CALL HEADIN ('ALLOWED TO VARY WITH RADIUS$',27,0.8,4)
      CALL MESSAG ('SPEC RMR =$',100,7.4,6.5)
      CALL REALNO (RMR(163),2,'ABUT','ABUT')
      CALL MESSAG('CASE 3$',100,8.7,6.0)
12    IF( K .NE. 4) GO TO 14
      CALL HEADIN('SOLIDITY,ADVANCE RATIO & TIP VELOCITY HELD CONSTANT$'
     *,51,0.8,4)
      CALL HEADIN (' CHORD & ROTATIONAL VELOCITY$',27,0.8,4)
      CALL HEADIN ('ALLOWED TO VARY WITH RADIUS$',27,0.8,4)
      CALL MESSAG (' SPEC RMR =$',100,7.4,6.5)
      CALL REALNO (RMR(163),2,'ABUT','ABUT')
      CALL MESSAG('CASE 4$',100,8.7,6.0)
14    CALL ENDPL(0)
      RETURN
      END
C*****************************************************************
C
C    SUBROUTINE TO CREATE GRAPH FOR ENDURANCE VELOCITY VS RADIUS
C
C*****************************************************************
C*****************************************************************
      SUBROUTINE GRAF2
      IMPLICIT REAL (A-H,L-Z)
       COMMON /C/ RMR(400),I,J,K,DRMR(400),MINX,MAXX,MINY,MAXY,MINZ,
     * MAXZ,DMIN,DMAX,MINB,MAXB,MINC,MAXC,MIN,MAX,MAXA,IANS,TEXT(200)
      COMMON /W/ RCMAX(400),DRCMAX(400),VFRAN(400),DVFRAN(400),
     * HOVALT(400),DHOVAL(400),SEVALT(400),DSEVAL(400),FMERT(400),
     * DFMERT(400),VFEND(400),DVFEND(400)
      CALL COMPRS
      CALL PAGE (11.0,8.5)
      CALL PHYSOR(1.5,1.5)
      CALL AREA2D (2.50,4.50)
      CALL XNAME ('RADIUS OF MAIN ROTOR (FT)$',100)
      CALL YNAME ('ENDURANCE VELOCITY (KNOTS)$',100)
      CALL GRAF (MINZ,'SCALE',MAXZ,MIN,'SCALE',MAX)
      CALL CURVE (RMR,VFEND,324,0)
```

```
      CALL DOT                                                          PER13930
      CALL GRID (1,2)                                                   PER13940
      CALL RESET('DOT')                                                 PER13950
      CALL ENDGR(0)                                                     PER13960
C-----------------------------------------------------------------------PER13970
      CALL PHYSOR(5.5,1.5)                                              PER13980
      CALL AREA2D (2.50,4.50)                                           PER13990
      CALL XNAME ('RADIUS OF MAIN ROTOR (% CHANGE)$',100)               PER14000
      CALL YNAME ('ENDURANCE VELOCITY (% CHANGE)$',100)                 PER14010
      CALL GRAF (DMIN,'SCALE',DMAX,MINY,'SCALE',MAXY)                   PER14020
      CALL CURVE (DRMR,DVFEND,324,0)                                    PER14030
      CALL DOT                                                          PER14040
      CALL GRID (1,2)                                                   PER14050
      CALL ENDGR(0)                                                     PER14060
C-----------------------------------------------------------------------PER14070
      CALL PHYSOR(1.5,1.5)                                              PER14080
      CALL AREA2D (6.50,4.70)                                           PER14090
      CALL HEADIN ('ENDURANCE VELOCITY VERSUS RADIUS$',32,1.4,3)        PER14100
      IF(K .NE. 1 ) GO TO 8                                             PER14110
      CALL HEADIN ('CHORD & ROTATIONAL VELOCITY HELD CONSTANT$',41,     PER14120
     *            0.8,3)                                                PER14130
      CALL HEADIN ('SOLIDITY, TIP VELOCITY & ADVANCE RATIO$',38,0.8,4)  PER14140
      CALL HEADIN('ALLOWED TO VARY WITH RADIUS$',27,0.8,4)              PER14150
      CALL MESSAG ('SPEC RMR =$',100,7.4,6.5)                           PER14160
      CALL REALNO (RMR(163),2,'ABUT','ABUT')                            PER14170
      CALL MESSAG('CASE 1$',100,8.7,6.0)                                PER14180
8     IF(K .NE. 2) GO TO 10                                             PER14190
      CALL HEADIN ('CHORD,TIP VELOCITY & ADVANCE RATIO HELD CONSTANT$', PER14200
     *       48,0.8,4)                                                  PER14210
      CALL HEADIN ('SOLIDITY, ROTATIONAL VELOCITY$',29,0.8,4)           PER14220
      CALL HEADIN ('ALLOWED TO VARY WITH RADIUS$',27,0.8,4)             PER14230
      CALL MESSAG ('SPEC RMR =$',100,7.4,6.5)                           PER14240
      CALL REALNO (RMR(163),2,'ABUT','ABUT')                            PER14250
      CALL MESSAG('CASE 2$',100,8.7,6.0)                                PER14260
10    IF( K .NE. 3) GO TO 12                                            PER14270
      CALL HEADIN (' SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT$',    PER14280
     *       44,0.8,4)                                                  PER14290
      CALL HEADIN (' CHORD,ADVANCE RATIO & TIP VELOCITY$',34, 0.8, 4)   PER14300
      CALL HEADIN ('ALLOWED TO VARY WITH RADIUS$',27,0.8,4)             PER14310
      CALL MESSAG ('SPEC RMR =$',100,7.4,6.5)                           PER14320
      CALL REALNO (RMR(163),2,'ABUT','ABUT')                            PER14330
      CALL MESSAG('CASE 3$',100,8.7,6.0)                                PER14340
12    IF( K .NE. 4) GO TO 14                                            PER14350
      CALL HEADIN('SOLIDITY,ADVANCE RATIO & TIP VELOCITY HELD CONSTANT$'PER14360
     *,51,0.8,4)                                                        PER14370
      CALL HEADIN (' CHORD & ROTATIONAL VELOCITY$',27,0.8,4)            PER14380
      CALL HEADIN ('ALLOWED TO VARY WITH RADIUS$',27,0.8,4)             PER14390
      CALL MESSAG ('SPEC RMR =$',100,7.4,6.5)                           PER14400
```

```
      CALL REALNO (RMR(163),2,'ABUT','ABUT')
      CALL MESSAG('CASE 4$',100,8.7,6.0)
14    CALL ENDPL(0)
      RETURN
      END
C***********************************************************************
C
C     SUBROUTINE TO CREATE GRAPH FOR MAXIMUM RATE OF CLIMB VS RADIUS
C
C***********************************************************************
C***********************************************************************
      SUBROUTINE GRAF3
      IMPLICIT REAL (A-H,L-Z)
       COMMON /C/ RMR(400),I,J,K,DRMR(400),MINX,MAXX,MINY,MAXY,MINZ,
     * MAXZ,DMIN,DMAX,MINB,MAXB,MINC,MAXC,MIN,MAX,MAXA,IANS,TEXT(200)
      COMMON /W/ RCMAX(400),DRCMAX(400),VFRAN(400),DVFRAN(400),
     * HOVALT(400),DHOVAL(400),SEVALT(400),DSEVAL(400),FMERT(400),
     * DFMERT(400),VFEND(400),DVFEND(400)
      CALL COMPRS
      CALL PAGE (11.0,8.5)
      CALL PHYSOR(1.5,1.5)
      CALL AREA2D (2.50,4.50)
      CALL XNAME ('RADIUS OF MAIN ROTOR (FT)$',100)
      CALL YNAME ('MAXIMUM RATE OF CLIMB (FT/SEC)$',100)
      CALL GRAF (MINZ,'SCALE',MAXZ,MIN,'SCALE',MAX)
      CALL CURVE (RMR,RCMAX,324,0)
      CALL DOT
      CALL GRID (1,2)
      CALL RESET('DOT')
      CALL ENDGR(0)
C-----------------------------------------------------------------------
      CALL PHYSOR(5.5,1.50)
      CALL AREA2D (2.50,4.50)
      CALL XNAME ('RADIUS OF MAIN ROTOR (% CHANGE)$',100)
      CALL YNAME ('MAXIMUM RATE OF CLIMB (% CHANGE)$',100)
      CALL GRAF (DMIN,'SCALE',DMAX,MINY,'SCALE',MAXY)
      CALL CURVE (DRMR,DRCMAX,324,0)
      CALL DOT
      CALL GRID (1,2)
      CALL ENDGR(0)
C-----------------------------------------------------------------------
      CALL PHYSOR(1.5,1.5)
      CALL AREA2D (6.50,4.70)
      CALL HEADIN ('MAXIMUM RATE OF CLIMB VERSUS RADIUS CHANGE$',
     *   42,1.4,4)
      IF(K .NE. 1 ) GO TO 8
      CALL HEADIN ('CHORD & ROTATIONAL VELOCITY HELD CONSTANT$',41,
     *              0.8,3)
```

```
      CALL HEADIN ('SOLIDITY, TIP VELOCITY & ADVANCE RATIO$',38,0.8,4)   PER14890
      CALL HEADIN('ALLOWED TO VARY WITH RADIUS$',27,0.8,4)               PER14900
      CALL MESSAG ('SPEC RMR =$',100,7.4,6.5)                            PER14910
      CALL REALNO (RMR(163),2,'ABUT','ABUT')                             PER14920
      CALL MESSAG('CASE 1$',100,8.7,6.0)                                 PER14930
8     IF(K .NE. 2) GO TO 10                                              PER14940
      CALL HEADIN ('CHORD,TIP VELOCITY & ADVANCE RATIO HELD CONSTANT$',  PER14950
     *       46,0.8,4)                                                   PER14960
      CALL HEADIN ('SOLIDITY, ROTATIONAL VELOCITY$',29,0.8,4)            PER14970
      CALL HEADIN ('ALLOWED TO VARY WITH RADIUS$',27,0.8,4)              PER14980
      CALL MESSAG ('SPEC RMR =$',100,7.4,6.5)                            PER14990
      CALL REALNO (RMR(163),2,'ABUT','ABUT')                             PER15000
      CALL MESSAG('CASE 2$',100,8.7,6.0)                                 PER15010
10    IF( K .NE. 3) GO TO 12                                             PER15020
      CALL HEADIN ('SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT$',      PER15030
     *       44,0.8,4)                                                   PER15040
      CALL HEADIN ('CHORD,ADVANCE RATIO & TIP VELOCITY$',34,0.8,4)       PER15050
      CALL HEADIN ('ALLOWED TO VARY WITH RADIUS$',27,0.8,4)              PER15060
      CALL MESSAG ('SPEC RMR =$',100,7.4,6.5)                            PER15070
      CALL REALNO (RMR(163),2,'ABUT','ABUT')                             PER15080
      CALL MESSAG('CASE 3$',100,8.7,6.0)                                 PER15090
12    IF( K .NE. 4) GO TO 14                                             PER15100
      CALL HEADIN('SOLIDITY,ADVANCE RATIO & TIP VELOCITY HELD CONSTANT$'PER15110
     *,51,0.8,4)                                                         PER15120
      CALL HEADIN ('CHORD & ROTATIONAL VELOCITY$',27,0.8,4)              PER15130
      CALL HEADIN ('ALLOWED TO VARY WITH RADIUS$',27,0.8,4)              PER15140
      CALL MESSAG ('SPEC RMR =$',100,7.4,6.5)                            PER15150
      CALL REALNO (RMR(163),2,'ABUT','ABUT')                             PER15160
      CALL MESSAG('CASE 4$',100,8.7,6.0)                                 PER15170
14    CALL ENDPL(0)                                                      PER15180
      RETURN                                                             PER15190
      END                                                                PER15200
C*********************************************************************** PER15210
C                                                                        PER15220
C     SUBROUTINE TO CREATE GRAPH FOR MAXIMUM RANGE VELOCITY VS RADIUS    PER15230
C                                                                        PER15240
C***********************************************************************PER15250
C***********************************************************************PER15260
      SUBROUTINE GRAF4                                                   PER15270
      IMPLICIT REAL (A-H,L-Z)                                            PER15280
       COMMON /C/ RMR(400),I,J,K,DRMR(400),MINX,MAXX,MINY,MAXY,MINZ,     PER15290
     * MAXZ,CMIN,CMAX,MINB,MAXB,MINC,MAXC,MIN,MAX,MAXA,IANS,TEXT(200)    PER15300
      COMMON /W/ RCMAX(400),DRCMAX(400),VFRAN(400),DVFRAN(400),          PER15310
     * HOVALT(400),DHOVAL(400),SEVALT(400),DSEVAL(400),FMERT(400),       PER15320
     * DFMERT(400),VFEND(400),DVFEND(400)                                PER15330
      CALL COMPRS                                                        PER15340
      CALL PAGE (11.0,8.5)                                               PER15350
      CALL PHYSOR(1.5,1.5)                                               PER15360
```

```
      CALL AREA2C (2.50,4.50)
      CALL XNAME ('RADIUS OF MAIN ROTOR (FT)$',100)
      CALL YNAME ('MAXIMUM RANGE VELOCITY (KNOTS)$',100)
      CALL GRAF (MINZ,'SCALE',MAXZ,MIN,'SCALE',MAX)
      CALL CURVE (RMR,VFRAN,324,-1 )
      CALL DOT
      CALL GRID (1,2)
      CALL RESET('DOT')
      CALL ENDGR(0)
C***********************************************************************
      CALL PHYSOR(5.5,1.5)
      CALL AREA2C (2.5,4.5)
      CALL XNAME ('RADIUS OF MAIN ROTOR (% CHANGE)$',100)
      CALL YNAME ('MAXIMUM RANGE VELOCITY (% CHANGE)$',100)
      CALL GRAF (DMIN,'SCALE',DMAX,MINY,'SCALE',MAXY)
      CALL CURVE (DRMR,DVFRAN,324,-1)
      CALL DOT
      CALL GRID (1,2)
      CALL ENDGR(0)
C-----------------------------------------------------------------------
      CALL PHYSOR(1.5,1.5)
      CALL AREA2C (6.50,4.70)
      CALL HEADIN ('MAXIMUM RANGE VELOCITY VERSUS RADIUS CHANGE$',
     *  43,1.4,4)
      IF(K .NE. 1 ) GO TO 8
      CALL HEADIN ('CHORD & ROTATIONAL VELOCITY HELD CONSTANT$',41,
     *           0.8,3)
      CALL HEADIN (' SOLIDITY, TIP VELOCITY & ADVANCE RATIO$',38,0.8,4)
      CALL HEADIN('ALLOWED TO VARY WITH RADIUS$',27,0.8,4)
      CALL MESSAG ('SPEC RMR =$',100,7.4,6.5)
      CALL REALNO (RMR(163),2,'ABUT','ABUT')
      CALL MESSAG('CASE 1$',100,8.7,6.0)
8     IF(K .NE. 2) GO TO 10
      CALL HEADIN ('CHORD,TIP VELOCITY & ADVANCE RATIO HELD CONSTANT$',
     *        48,0.8,4)
      CALL HEADIN (' SOLIDITY, ROTATIONAL VELOCITY$',29,0.8,4)
      CALL HEADIN ('ALLOWED TO VARY WITH RADIUS$',27,0.8,4)
      CALL MESSAG ('SPEC RMR =$',100,7.4,6.5)
      CALL REALNO (RMR(163),2,'ABUT','ABUT')
      CALL MESSAG('CASE 2$',100,8.7,6.0)
10    IF( K .NE. 3) GO TO 12
      CALL HEADIN (' SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT$',
     *        44,0.8,4)
      CALL HEADIN ('CHORD,ADVANCE RATIO & TIP VELOCITY$',34,0.8,4)
      CALL HEADIN ('ALLOWED TO VARY WITH RADIUS$',27,0.8,4)
      CALL MESSAG ('SPEC RMR =$',100,7.4,6.5)
      CALL REALNO (RMR(163),2,'ABUT','ABUT')
      CALL MESSAG('CASE 3$',100,8.7,6.0)
```

```
12    IF( K .NE. 4) GO TO 14
      CALL HEADIN('SOLIDITY,ADVANCE RATIO & TIP VELOCITY HELD CONSTANT$'
     *,51,0.8,4)
      CALL HEADIN ('CHORD & ROTATIONAL VELOCITY$',27,0.8,4)
      CALL HEADIN ('ALLOWED TO VARY WITH RADIUS$',27,0.8,4)
      CALL MESSAG ('SPEC RMR =$',100,7.4,6.5)
      CALL REALNO (RMR(163),2,'ABUT','ABUT')
      CALL MESSAG('CASE 4$',100,8.7,6.0)
14    CALL ENDPL(0)
      RETURN
      END
C*******************************************************************
C
C     SUBROUTINE TO CREATE GRAPH FOR MAXIMUM HOVER CEILING
C
C********************************************************************
C********************************************************************
      SUBROUTINE GRAF5
      IMPLICIT REAL (A-H,L-Z)
       COMMON /C/ RMR(400),I,J,K,DRMR(400),MINX,MAXX,MINY,MAXY,MINZ,
     * MAXZ,DMIN,DMAX,MINB,MAXB,MINC,MAXC,MIN,MAX,MAXA,IANS,TEXT(200)
      COMMON /W/ RCMAX(400),DRCMAX(400),VFRAN(400),DVFRAN(400),
     * HOVALT(400),DHOVAL(400),SEVALT(400),DSEVAL(400),FMERT(400),
     * DFMERT(400),VFEND(400),DVFEND(400)
      CALL COMPRS
      CALL PAGE (11.0,8.5)
      CALL PHYSOR(1.5,1.5)
      CALL AREA2D (2.50,4.50)
      CALL XNAME ('RADIUS OF MAIN ROTOR (FT)$',100)
      CALL YNAME ('MAXIMUM HOVER CEILING (FT)$',100)
      CALL GRAF (MINZ,'SCALE',MAXZ,MIN,'SCALE',MAX)
      CALL CURVE (RMR,HOVALT,324,0)
      CALL DOT
      CALL GRID (1,2)
      CALL RESET('DOT')
      CALL ENDGR(0)
C-------------------------------------------------------------------
      CALL PHYSOR(5.5,1.50)
      CALL AREA2D (2.50,4.50)
      CALL XNAME ('RADIUS OF MAIN ROTOR (% CHANGE)$',100)
      CALL YNAME ('MAXIMUM HOVER CEILING (% CHANGE)$',100)
      CALL GRAF (DMIN,'SCALE',DMAX,MINY,'SCALE',MAXY)
      CALL CURVE (DRMR,DHOVAL,324,0)
      CALL DOT
      CALL GRID (1,2)
      CALL ENDGR(0)
C-------------------------------------------------------------------
      CALL PHYSOR(1.5,1.5)
```

```
      CALL AREA2D (6.50,4.70)                                           PER16330
      CALL HEADIN ('MAXIMUM HOVER CEILING (IGE) VERSUS RADIUS$',        PER16340
     *  41,1.2,4)                                                       PER16350
      IF(K .NE. 1 ) GO TO 8                                             PER16360
      CALL HEADIN ('CHORD & ROTATIONAL VELOCITY HELD CONSTANT$',41,     PER16370
     *            0.8,3)                                                PER16380
      CALL HEADIN ('SOLIDITY, TIP VELOCITY & ADVANCE RATIO$',38,0.8,4)  PER16390
      CALL HEADIN('ALLOWED TO VARY WITH RADIUS$',27,0.8, 4)             PER16400
      CALL MESSAG ('SPEC RMR =$',100,7.4,6.5)                           PER16410
      CALL REALNO (RMR(163),2,'ABUT','ABUT')                            PER16420
      CALL MESSAG('CASE 1$',100,8.7,6.0)                                PER16430
8     IF(K .NE. 2) GO TO 10                                             PER16440
      CALL HEADIN ('CHORD,TIP VELOCITY & ADVANCE RATIO HELD CONSTANT$', PER16450
     *    48,0.8,4)                                                     PER16460
      CALL HEADIN ('SOLIDITY, ROTATIONAL VELOCITY$',29,0.8,4)           PER16470
      CALL HEADIN (' ALLOWED TO VARY WITH RADIUS$',27,0.8,4)            PER16480
      CALL MESSAG ('SPEC RMR =$',100,7.4,6.5)                           PER16490
      CALL REALNO (RMR(163),2,'ABUT','ABUT')                            PER16500
      CALL MESSAG('CASE 2$',100,8.7,6.0)                                PER16510
10    IF( K .NE. 3) GO TO 12                                            PER16520
      CALL HEADIN ('SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT$',     PER16530
     *    44,0.8,4)                                                     PER16540
      CALL HEADIN ('CHORD,ADVANCE RATIO & TIP VELOCITY$',34,0.8,4)      PER16550
      CALL HEADIN (' ALLOWED TO VARY WITH RADIUS$',27,0.8,4)            PER16560
      CALL MESSAG ('SPEC RMR =$',100,7.4,6.5)                           PER16570
      CALL REALNO (RMR(163),2,'ABUT','ABUT')                            PER16580
      CALL MESSAG('CASE 3$',100,8.7,6.0)                                PER16590
12    IF( K .NE. 4) GO TO 14                                            PER16600
      CALL HEADIN('SOLIDITY,ADVANCE RATIO & TIP VELOCITY HELD CONSTANT$'PER16610
     *,51,0.8,4)                                                        PER16620
      CALL HEADIN ('CHORD & ROTATIONAL VELOCITY$',27,0.8,4)             PER16630
      CALL HEADIN (' ALLOWED TO VARY WITH RADIUS$',27,0.8,4)            PER16640
      CALL MESSAG ('SPEC RMR =$',100,7.4,6.5)                           PER16650
      CALL REALNO (RMR(163),2,'ABUT','ABUT')                            PER16660
      CALL MESSAG('CASE 4$',100,8.7,6.0)                                PER16670
14    CALL ENDPL(0)                                                     PER16680
      RETURN                                                            PER16690
      END                                                               PER16700
C*********************************************************************** PER16710
C                                                                       PER16720
C     SUBROUTINE TO CREATE GRAPH FOR MAXIMUM SERVIVE CEILING            PER16730
C                                                                       PER16740
C***********************************************************************PER16750
C***********************************************************************PER16760
      SUBROUTINE GRAF6                                                  PER16770
      IMPLICIT REAL (A-H,L-Z)                                           PER16780
       COMMON /C/ RMR(400),I,J,K,DRMR(400),MINX,MAXX,MINY,MAXY,MINZ,    PER16790
     * MAXZ,CMIN,CMAX,MINB,MAXB,MINC,MAXC,MIN,MAX,MAXA,IANS, TEXT(200)  PER16800
```

```
      COMMON /W/ RCMAX(400),DRCMAX(400),VFRAN(400),DVFRAN(400),
     * HOVALT(400),DHOVAL(400),SEVALT(400),DSEVAL(400),FMERT(400),
     * DFMERT(400),VFEND(400),DVFEND(400)
      CALL COMPRS
      CALL PAGE (11.0,8.5)
      CALL PHYSOR(1.5,1.5)
      CALL AREA2D (2.50,4.50)
      CALL XNAME ('RADIUS OF MAIN ROTOR (FT)$',100)
      CALL YNAME ('MAXIMUM SERVICE CEILING (FT)$',100)
      CALL GRAF (MINZ,'SCALE',MAXZ,MIN,'SCALE',MAX)
      CALL CURVE (RMR,SEVALT,322,0)
      CALL DOT
      CALL GRID (1,2)
      CALL RESET('DOT')
      CALL ENDGR(0)
C-----------------------------------------------------------------------
      CALL PHYSOR(5.5,1.5)
      CALL AREA2D (2.50,4.50)
      CALL XNAME ('RADIUS OF MAIN ROTOR (% CHANGE)$',100)
      CALL YNAME ('MAXIMUM SERVICE CEILING (% CHANGE)$',100)
      CALL GRAF (DMIN,'SCALE',DMAX,MINY,'SCALE',MAXY)
      CALL CURVE (DRMR,DSEVAL,324,0)
      CALL DOT
      CALL GRID (1,2)
      CALL ENDGR(0)
C-----------------------------------------------------------------------
      CALL PHYSOR(1.5,1.5)
      CALL AREA2D (6.50,4.70)
      CALL HEADIN ('MAXIMUM SERVICE CEILING VERSUS RADIUS CHANGE$',
     *    44,1.4,4)
      IF(K .NE. 1 ) GO TO 8
      CALL HEADIN ('CHORD & ROTATIONAL VELOCITY HELD CONSTANT$',41,
     *             0.8,3)
      CALL HEADIN ('SOLIDITY, TIP VELOCITY & ADVANCE RATIO$',38,0.8,4)
      CALL HEADIN('ALLOWED TO VARY WITH RADIUS$',27,0.8,4)
      CALL MESSAG ('SPEC RMR =$',100,7.4,6.5)
      CALL REALNO (RMR(163),2,'ABUT','ABUT')
      CALL MESSAG('CASE 1$',100,8.7,6.0)
8     IF(K .NE. 2) GO TO 10
      CALL HEADIN ('CHORD,TIP VELOCITY & ADVANCE RATIO HELD CONSTANT$',
     *    48,0.8,4)
      CALL HEADIN ('SOLIDITY, ROTATIONAL VELOCITY$',29,0.8,4)
      CALL HEADIN ('ALLOWED TO VARY WITH RADIUS$',27,0.8,4)
      CALL MESSAG ('SPEC RMR =$',100,7.4,6.5)
      CALL REALNO (RMR(163),2,'ABUT','ABUT')
      CALL MESSAG('CASE 2$',100,8.7,6.0)
10    IF( K .NE. 3) GO TO 12
      CALL HEADIN ('SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT$',
```

```
     *      44,C.8,4)
      CALL HEADIN ('CHORD,ADVANCE RATIO & TIP VELCCITY$',34,0.8,4)
      CALL HEADIN ('ALLOWED TO VARY WITH RADIUS$',27,0.8,4)
      CALL MESSAG ('SPEC RMR =$',100,7.4,6.5)
      CALL REALNO (RMR(163),2,'ABUT','ABUT')
      CALL MESSAG('CASE 3$',100,8.7,6.0)
 12   IF( K .NE. 4) GO TO 14
      CALL HEADIN('SOLIDITY,ADVANCE RATIC & TIP VELOCITY HELD CONSTANT$'
     *,51,0.8,4)
      CALL HEADIN ('CHORD & ROTATIONAL VELCCITY$',27,0.8,4)
      CALL HEADIN ('ALLOWED TO VARY WITH RADIUS$',27,0.8,4)
      CALL MESSAG ('SPEC RMR =$',100,7.4,6.5)
      CALL REALNO (RMR(163),2,'ABUT','ABUT')
      CALL MESSAG('CASE 4$',100,8.7,6.0)
 14   CALL ENDPL(0)
      RETURN
      END
C***********************************************************************
C
C     SUBROUTINE TO CREATE GRAPH OF FIGURE OF MERIT VS RADIUS
C
C***********************************************************************
C***********************************************************************
      SUBROUTINE GRAF1E
      IMPLICIT REAL (A-H,L-Z)
       COMMON /C/ RMR(400),I,J,K,DRMR(400),MINX,MAXX,MINY,MAXY,MINZ,
     * MAXZ,CMIN,CMAX,MINB,MAXB,MINC,MAXC,MIN,MAX,MAXA,IANS,TEXT(200)
      COMMON /W/ RCMAX(400),DRCMAX(400),VFRAN(400),DVFRAN(400),
     * HOVALT(400),DHOVAL(400),SEVALT(400),DSEVAL(400),FMERT(400),
     * DFMERT(400),VFEND(400),DVFEND(400)
      CALL COMPRS
      CALL PAGE (11.0,8.5)
      CALL PHYSOR(1.5,1.5)
      CALL AREA2D (2.50,4.50)
      CALL XNAME ('RADIUS OF MAIN ROTOR (FT)$',100)
      CALL YNAME ('FIGUURE OF MERIT$',100)
      CALL GRAF (MINZ,'SCALE',MAXZ,MINB,'SCALE',MAXB)
      CALL CURVE (RMR,FMERT,324,0)
      CALL DOT
      CALL GRID (1,2)
      CALL RESET('DOT')
      CALL ENDGR(0)
C-----------------------------------------------------------------------
      CALL PHYSOR(5.5,1.5)
      CALL AREA2D (2.50,4.50)
      CALL XNAME ('RADIUS OF MAIN ROTOR (% CHANGE)$',100)
      CALL YNAME ('FIGURE OF MERIT (% CHANGE)$',100)
      CALL GRAF (DMIN,'SCALE',DMAX,MINC,'SCALE',MAXC)
```

```
      CALL CURVE (DRMR,DFMERT,324,0)                                    PER17770
      CALL DOT                                                          PER17780
      CALL GRID (1,2)                                                   PER17790
      CALL ENDGR(0)                                                     PER17800
C-----------------------------------------------------------------------PER17810
      CALL PHYSOR(1.5,1.5)                                              PER17820
      CALL AREA2D (6.50,4.70)                                           PER17830
      CALL HEADIN ('FIGURE OF MERIT VERSUS RADIUS CHANGE$',             PER17840
     *    36,1.4,4)                                                     PER17850
      IF(K .NE. 1 ) GO TO 8                                             PER17860
      CALL HEADIN (' CHORD & ROTATIONAL VELOCITY HELD CONSTANT$',41,    PER17870
     *             0.8,3)                                               PER17880
      CALL HEADIN ('SOLIDITY, TIP VELOCITY & ADVANCE RATIO$',38,0.8,4)  PER17890
      CALL HEADIN('ALLOWED TO VARY WITH RADIUS$',27,0.8, 4)             PER17900
      CALL MESSAG (' SPEC RMR =$',100,7.4,6.5)                          PER17910
      CALL REALNO (RMR(163),2,'ABUT','ABUT')                            PER17920
      CALL MESSAG('CASE 1$',100,8.7,6.0)                                PER17930
8     IF(K .NE. 2) GO TO 10                                             PER17940
      CALL HEADIN (' CHORD,TIP VELOCITY & ADVANCE RATIO HELD CONSTANT$', PER17950
     *       48,0.8,4)                                                  PER17960
      CALL HEADIN (' SOLIDITY, ROTATIONAL VELOCITY$',29,0.8,4)          PER17970
      CALL HEADIN (' ALLOWED TO VARY WITH RADIUS$',27,0.8,4)            PER17980
      CALL MESSAG (' SPEC RMR =$',100,7.4,6.5)                          PER17990
      CALL REALNO (RMR(163),2,'ABUT','ABUT')                            PER18000
      CALL MESSAG('CASE 2$',100,8.7,6.0)                                PER18010
10    IF( K .NE. 3) GO TO 12                                            PER18020
      CALL HEADIN (' SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT$',    PER18030
     *       44,0.8,4)                                                  PER18040
      CALL HEADIN (' CHORD,ADVANCE RATIO & TIP VELOCITY$',34,0.8,4)     PER18050
      CALL HEADIN (' ALLOWED TO VARY WITH RADIUS$',27,0.8,4)            PER18060
      CALL MESSAG (' SPEC RMR =$',100,7.4,6.5)                          PER18070
      CALL REALNO (RMR(163),2,'ABUT','ABUT')                            PER18080
      CALL MESSAG('CASE 3$',100,8.7,6.0)                                PER18090
12    IF( K .NE. 4) GO TO 14                                            PER18100
      CALL HEADIN('SOLIDITY,ADVANCE RATIO & TIP VELOCITY HELD CONSTANT$'PER18110
     *,51,0.8,4)                                                        PER18120
      CALL HEADIN (' CHORD & ROTATIONAL VELOCITY$',27,0.8,4)            PER18130
      CALL HEADIN (' ALLOWED TO VARY WITH RADIUS$',27,0.8,4)            PER18140
      CALL MESSAG (' SPEC RMR =$',100,7.4,6.5)                          PER18150
      CALL REALNO (RMR(163),2,'ABUT','ABUT')                            PER18160
      CALL MESSAG('CASE 4$',100,8.7,6.0)                                PER18170
14    CALL ENDPL(0)                                                     PER18180
      RETURN                                                            PER18190
      END                                                               PER18200
C=======================================================================PER18210
C                                                                       PER18220
C   SUBROUTINE RDINT -- INTERACTIVELY READS AN INTEGER REPLY           =PER18230
C   INTO A FORTRAN PROGRAM.  IF THE USER INADVERTENLY ENTERS AN IMPROPER=PER18240
```

```
C   DATA CHARACTER THE S/R ISSUES A WARNING AND ALLOWS A RECOVERY.       =PER18250
C                                                                          PER18260
C========================================================================PER18270
      SUBROUTINE RDINT (IANS)                                              PER18280
      INTEGER COUNT                                                        PER18290
C------------------------------------------------------------------------PER18300
      COUNT=0                                                              PER18310
10    CONTINUE                                                             PER18320
      COUNT=COUNT+1                                                        PER18330
      IF (COUNT.LT.3) GO TO 20                                             PER18340
      WRITE (5,60)                                                         PER18350
      GO TO 50                                                             PER18360
20    CONTINUE                                                             PER18370
      READ (5,*,END=40,ERR=40) IANS                                        PER18380
      IF (IANS) 40,40,30                                                   PER18390
30    CONTINUE                                                             PER18400
      RETURN                                                               PER18410
40    REWIND 5                                                             PER18420
      WRITE (5,70)                                                         PER18430
      GO TO 10                                                             PER18440
50    CONTINUE                                                             PER18450
      STOP                                                                 PER18460
C------------------------------------------------------------------------PER18470
60    FORMAT (//,5X,49HPROGRAM TERMINATION - TWO IMPROPER DATA ENTRIES     PER18480
     1)                                                                    PER18490
70    FORMAT (1X,56HWARNING: IMPROPER DATA ENTRY   ENTER A POSITIVE INTEPER18500
     1GER.)                                                                PER18510
      END                                                                  PER18520
```

LIST OF REFERENCES

1. Integrated Software Systems Corporation, DISSPLA : Pocket Guide, Version 9.0, 1981.

2. Layton, Professor Donald M., Helicopter Performance, Naval Postgraduate School, Monterey California, 1983.

3. Layton, Professor Donald M., AE 4306 Design Design Manual, Naval Postgraduate School, Monterey California, 1983.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2.	Army Aviation Systems Command Attn: Mr Crawford DRAV-GT 4300 Goodfellow Blvd St. Louis, Missouri 63120	1
3.	Library, Code 0142 Naval Postgraduate School Monterey, California 93943	2
4.	Department Chairman, Code 67 Department of Aeronautics Naval Postgraduate School Monterey, California 93943	1
5.	Prof Donald M. Layton, Code 67-LN Department of Aeronautics Naval Postgraduate School Monterey, California 93943	5
6.	CPT Patrick V. Adamcik c/o CPT Steve Kee 302 Lakeshore Road Enterprise, Alabama 36330	4

1 3 37 5

209712

Thesis
A222 Adamcik
c.1 Preliminary helicopter design decision making based on flight performance factors.